



TUGAS AKHIR - TF 145565

**RANCANG BANGUN SISTEM MONITORING KECEPATAN
ANGIN BERBASIS MIKROKONTROLER ATMEGA32
PADA TURBIN ANGIN TIPE HORIZONTAL AXIS**

SINTA DEWI RAHMAWATI
NRP. 2413 031 079

Dosen Pembimbing
Andi Rahmadiansah ST, MT
NIP. 19790517 200312 1 002

PROGRAM STUDI D3 METROLOGI DAN INSTRUMENTASI
JURUSAN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



TUGAS AKHIR - TF 145565

***WIND SPEED MONITORING SYSTEM DESIGN WITH
MICROCONTROLLER ATMEGA32 ON WIND TURBINE
HORIZONTAL AXIS***

SINTA DEWI RAHMAWATI
NRP. 2413 031 079

Advisor Lecturer
Andi Rahmadiansah ST, MT
NIP. 19790517 200312 1 002

Study Program Of D3 Metrology And Instrumentation
Department of Engineering Physics
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2016

**RANCANG BANGUN SISTEM MONITORING
KECEPATAN ANGIN BERBASIS
MIKROKONTROLER ATMEGA32 PADA TURBIN
ANGIN TIPE HORIZONTAL AXIS**

TUGAS AKHIR

Oleh :

SINTA DEWI RAHMAWATI

NRP. 2413 031 079

Surabaya, Juli 2016

Mengetahui / Menyetujui

Dosen Pembimbing



Andi Rahmadiansah, ST, MT


NIP. 19790517 200312 1 002

**Ketua Jurusan
Teknik Fisika FTI-ITS**



Agus Muhammad Hatta, ST, MSi, Ph.D
NIP. 19780902 200312 1 002

**Ketua Program Studi
DIII Metrologi dan Instrumentasi**



Dr. Ir. Purwadi Agus D, M.Sc
NIP. 19620822 198803 1 001

**RANCANG BANGUN SISTEM MONITORING
KECEPATAN ANGIN BERBASIS
MIKROKONTROLER ATMEGA32 PADA TURBIN
ANGIN TIPE HORIZONTAL AXIS**


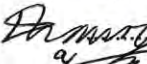


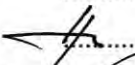
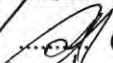

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Ahli Madya
pada
Program Studi D3 Metrologi dan Instrumentasi
Jurusan Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh :

SINTA DEWI RAHMAWATI
NRP. 2413 031 079

Disetujui oleh Tim Penguji Tugas Akhir :

1. Andi Rahmadiansah, S.T., M.T.  (Pembimbing)
2. Ir. Tutug Danardhono, M.T.  (Ketua Tim Penguji)
3. Ir. Harsono Hadi, M.Sc.Ph.D.  (Penguji I)
4. Hendra Cordova, S.T., M.T.  (Penguji II)
5. Ir. Roekmono, M.T.  (Penguji III)
6. Arief Abdurrahman, S.T., M.T.  (Penguji IV)
7. Murry Raditya, S.T., M.T.  (Penguji V)

**SURABAYA
JULI 2016**

RANCANG BANGUN SISTEM MONITORING KECEPATAN ANGIN BERBASIS MIKROKONTROLER ATMEGA32 PADA TURBIN ANGIN TIPE HORIZONTAL AXIS

Nama : Sinta Dewi Rahmawati
: Shita Agustin Kusumaningtyas
NRP : 2413031079
Jurusan : Prodi D3 Metrologi dan
Instrumentasi, Teknik Fisika
FTI-ITS
Dosen Pembimbing : Andi Rahmadiansah ST, MT

Lizda Johar M, S.T. M.T.

Abstrak

Sistem monitoring kecepatan angin pada turbin angin difungsikan untuk mendapatkan data kecepatan angin yang akurat, serta sebagai data acuan dalam menentukan lokasi pemasangan turbin angin. Oleh karena itu pada tugas akhir ini telah dirancang sistem pengukur kecepatan angin untuk mengetahui karakteristik alat ukur. Sistem monitoring ini menggunakan alat ukur kecepatan angin dengan modul optocoupler sebagai sensor yang berbasis mikrokontroler ATmega32. Pengujian dilakukan dengan membandingkan alat ukur kecepatan angin yang telah dibuat dengan alat ukur kecepatan angin standar. Dari hasil pengujian pada alat ukur diketahui alat ukur mempunyai ketidakpastian UA_1 sebesar 0,3012. Nilai ketidakpastian UA_2 sebesar 0,4638, UB_1 sebesar 0,0002886, ketidakpastian diperluas (Uexpand) sebesar 1,1644. Sehingga berdasarkan perhitungan ketidakpastian diperluas diatas, menghasilkan nilai ketidakpastian alat sebesar $\pm 1,164477453$. Untuk monitoring menggunakan Visual Studio Net 2013, MySQL, dan data logger menggunakan MMC. Hasil monitoring menunjukkan bahwa kecepatan angin yang terjadi tanggal 19 Juli 2016 pukul 22:09:23 – 22:19:53 memiliki rata-rata kecepatan 2,554 m/s, dan 25 Juli 2016 pukul 18:36:47 – 18:39:53 memiliki rata-rata kecepatan 8,409 m/s.

***Kata kunci: monitoring, alat ukur kecepatan angin, ketidakpastian
alat ukur, turbin angin tipe horizontal axis***

WIND SPEED MONITORING SYSTEM DESIGN WITH MICROCONTROLLER ATMEGA32 ON WIND TURBINE HORIZONTAL AXIS

Name : Sinta Dewi Rahmawati
: Shita Agustin Kusumaningtyas
NRP : 2413031079
Departement : Prodi D3 Metrologi dan
Instrumentasi, Teknik Fisika
FTI-ITS
Supervisor Lecture : Andi Rahmadiansah ST, MT

Abstract

Monitoring system wind speed in a wind turbine functioned to get data wind speeds are accurate, and as reference data in determining installment area wind turbines. Hence on their work end of this has been designed system measuring wind speed to know characteristic of a measuring instrument. Monitoring system is uses the gauges the wind with modules optocoupler as a sensor which based microcontroller ATmega32. Testing done by comparing the gauges the wind have been made with the gauges the wind standard. Form the our tests at a measuring instrument known a measuring instrument have uncertainty U_{a1} of 0.3012. Value of certainty U_{a2} 0.4638, U_{b1} of 0.0002886, uncertainty extended (U_{exp}) as much as 1.1644. So based on calculation un certainty expanded above, produce values uncertainty instrument of ± 1.164477453 . For monitoring use visual studio net 2013, MySQL, and data logger using MMC. The result of monitoring shows that wind speeds are happened 19 July 2016 at 22:09:23 – 22:19:53 having an average speed 2.554 m/s, and 25 July 2016 at 18:36:47 – 18:39:53 having an average speed 8.409 m/s.

Keywords: monitoring, wind speed measurement tool, uncertainty, the characteristics of the measuring instrument, Wind Turbine Horizontal Axis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
ABSTRAK	v
<i>ABSTRACT</i>	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii

BAB I PENDAHULUAN

1.1	Latar Belakang	1
1.2	Rumusan Masalah	3
1.3	Batasan Masalah	3
1.4	Tujuan	4
1.5	Metodologi Penelitian	4
1.6	Sistematika Laporan	5

BAB II TEORI PENUNJANG

2.1	Potensi Angin	7
2.2	Turbin Angin	9
2.3	<i>Horizontal Axis Wind Turbine</i>	9
2.4	Teori Turbin Angin	10
2.5	Anemometer	12
2.6	Sensor Optocoupler	14
2.7	Mikrokontroler ATmega8535	15
2.8	<i>LCD(Liquid Crystal Display) 16x2</i>	18
2.9	Karakteristik Alat Ukur	18
2.10	Ketidakpastian Kalibrasi	22
2.11	Microsoft Visual Studio 2013	25
2.12	MySQL	26
2.13	MMC (<i>Multi Media Card</i>)	27

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1	Diagram Alir (<i>flowchart</i>)	29
3.2	Keterangan <i>Flowchart</i>	30

BAB IV ANALISA DATA DAN PEMBAHASAN

4.1	Pengujian Alat	41
4.2	Analisa Data	56

BAB V PENUTUP

5.1	Kesimpulan	57
5.2	Saran	57

DAFTAR PUSTAKA

LAMPIRAN A

LAMPIRAN B

LAMPIRAN C

LAMPIRAN D

DAFTAR TABEL

Tabel 3.1	Parameter <i>Wind Turbine</i>	30
Tabel 4.1	Uji Tegangan Pada Mikrokontroler ATMega32	42
Tabel 4.2	Data Pengujian Alat	43
Tabel 4.3	Data Karakteristik Statik Alat	45
Tabel 4.4	Data Hubungan Antara Frekuensi ke Tegangan	45
Tabel 4.5	Database Kalibrasi	48
Tabel 4.6	Database Monitoring Kecepatan Angin pada tanggal 19 Juli 2016	52
Tabel 4.7	Database Monitoring Kecepatan Angin pada tanggal 25 Juli 2016	54

DAFTAR GAMBAR

Gambar 2.1	Anemometer Cup	13
Gambar 2.2	Modul Sensor Optocoupler	14
Gambar 2.3	Rangkaian Pengkondisian Sinyal LM393	15
Gambar 2.4	Blok diagram fungsional ATmega32	16
Gambar 2.5	Bentuk LCD (<i>Liquid Crystal Display</i>) 2x16	17
Gambar 2.6	Karakteristik output instrumen linier	19
Gambar 2.7	Grafik non-linieritas (Bentley,2005)	19
Gambar 2.8	Penjelasan Akurasi dan Presisi	21
Gambar 2.9	Konfigurasi MMC	26
Gambar 3.1	<i>Flowchart</i> Tugas Akhir	28
Gambar 3.2	Konstruksi turbin angin horizontal axis	30
Gambar 3.3	Diagram Blok Konstruksi Sensor Kecepatan Angin	33
Gambar 3.4	Rangkaian Modul <i>Sensor Optocoupler</i>	33
Gambar 3.5	Konstruksi <i>sensor optocoupler</i> dengan <i>disk rotary encoder</i>	33
Gambar 3.6	<i>Display Hardware</i> alat ukur kecepatan angin	34
Gambar 3.7	<i>Flowchart</i> Perancangan Sensor	35
Gambar 3.8	<i>Software</i> CVAVR	36
Gambar 3.9	Konfigurasi pin ATmega32	37
Gambar 3.10	Blok Integrasi Alat	38
Gambar 3.11	<i>Display Hyperterminal</i> Komunikasi Serial alat ukur kecepatan angin	38
Gambar 3.12	<i>Flowchart</i> Perancangan <i>Software Visual Studio</i>	39
Gambar 3.13	Tampilan Awal <i>Software</i> monitoring dengan <i>Visual Studio</i>	40
Gambar 4.1	<i>Minimum System</i> ATmega32	41

Gambar 4.2	Diagram Blok Modul Sensor Optocoupler	42
Gambar 4.3	Konfigurasi Modul Sensor Optocoupler	43
Gambar 4.4	Grafik Pembacaan Alat dan Pembacaan Standar	44
Gambar 4.5	Grafik Perbandingan Uji dengan Standar Anemometer	44
Gambar 4.6	Hubungan antara Frekuensi ke Tegangan	46
Gambar 4.7	<i>Histerisis</i>	48
Gambar 4.8	<i>Interface</i> Monitoring Kecepatan Angin	52
Gambar 4.9	Grafik Perbandingan kenaikan kecepatan angin terhadap waktu pada tanggal 19 Juli 2016	53
Gambar 4.10	Grafik Perbandingan Kenaikan Kecepatan Angin terhadap waktu pada tanggal 25 Juli 2016	54

BAB I

PENDAHULUAN

1.1 Latar Belakang

Energi merupakan suatu kebutuhan primer yang sangat dibutuhkan dalam kehidupan, energi memiliki peran yang dominan, yakni sebagai salah satu pendapatan Negara, bahan bakar dan bahan baku industri, penggerak kegiatan ekonomi, dan beberapa peran penting lainnya, dimana hampir semua aktivitas manusia berhubungan dengan energi. Namun kebutuhan energi tersebut tidak berbanding lurus dengan ketersediaan sumber energi. Di Indonesia bahan bakar fosil masih menjadi bahan bakar andalan dalam mencukupi kebutuhan pokok energi. Menurut Dewan Energi Nasional (DEN), ketergantungan terhadap energi fosil, terutama minyak bumi dalam pemenuhan konsumsi di dalam Negeri masih tinggi, yaitu sebesar 96% (minyak bumi 48%, gas 18%, dan batubara 30%) dari total konsumsi energi nasional. Dalam upaya pemenuhan kebutuhan energi tersebut mendorong semakin gencarnya dalam eksploitasi sumber bahan bakar fosil yang menyebabkan faktor dari kelangkaan bahan bakar fosil. Dari hal tersebut untuk menurunkan tingkat ketergantungan pada bahan bakar fosil dan untuk membantu dalam memasok energi tambahan dibutuhkan adanya sumber energi baru dan terbarukan.

Energi baru dan terbarukan mulai dikembangkan seiring dengan semakin menipisnya persediaan bahan bakar fosil dan juga untuk menanggulangi efek dari bahan bakar fosil tersebut. Menurut Peraturan Presiden Nomor 05 tahun 2006 tentang Kebijakan Energi Nasional (KEN) telah menetapkan target pemanfaatan energi baru dan terbarukan (EBT) sebesar 17% dari total Bauran Energi Nasional (BEN) pada tahun 2025. Kebijakan Energi Nasional (KEN) ditetapkan oleh pemerintah selain untuk mengatasi permasalahan persediaan bahan bakar fosil adalah untuk mengatasi wilayah yang belum terjangkau dengan energi, khususnya energi listrik pada wilayah perbatasan Indonesia. Rasio elektrifikasi Indonesia 81,70%, menunjukkan bahwa belum

seluruh daerah di Indonesia mendapatkan fasilitas listrik. terdapat kurang lebih 1.690 desa di Maluku dan Maluku Utara yang belum terjangkau energi listrik [5]. Total terdapat 890 desa di Maluku dan 800 di Maluku Utara, 400 desa diantaranya belum terjangkau listrik, dan 35 desa sudah terjangkau listrik melalui program listrik pulau terluar [7].

Dalam pengembangan energi baru dan terbarukan, angin atau energi bayu memiliki potensi yang cukup besar untuk dimanfaatkan menjadi energi listrik di Indonesia. Kondisi geografi Indonesia yang terdiri dari 2/3 wilayah lautan dan memiliki garis pantai ± 80.791 Km. Di Indonesia, kecepatan angin berkisar antara 2 m/s hingga 6 m/s. Dengan karakteristik kecepatan seperti itu, Indonesia dinilai cocok untuk menggunakan pembangkit listrik tenaga angin skala kecil (10 kW) dan menengah (10 – 100 kW); untuk penggunaan energi seperti misalnya, lampu, pompa air, alat – alat elektronik, dan lain-lain. Pusat tenaga angin sebagian besar masih berada di Nusa Tenggara Timur yang memiliki kecepatan angin lebih dari 5 m/s. ditunjukkan bahwa potensi angin di Indonesia mencapai 9,286 MW; dimana penggunaan hingga tahun 2004 masih kurang dari 0,5 MW berdasarkan data dari *Indonesia Energi Outlook and Statistics 2004*.

Energi angin dapat dimanfaatkan melalui penggunaan turbin angin. Ada beberapa tipe turbin angin yang digunakan saat ini, salah satunya turbin angin tipe horizontal axis. Turbin angin tipe ini merupakan salah satu turbin angin yang cocok untuk daerah dengan potensi energi angin berkecepatan rendah, dengan efisien tinggi, rendah dalam vibrasi dan bising, serta memiliki potensial power yang tinggi dari pada tipe turbin angin VAWT. Di wilayah indonesia, khususnya di Tanjung Perak Surabaya memiliki kecepatan angin laut antara 2 – 10 knot [2]. Dengan kondisi demikian maka turbin angin untuk skala kecil dapat beroperasi dengan baik, khususnya dalam instalasi turbin angin untuk energi alternatif pada pesisir pantai. Menurut data hasil penelitian mengenai pemasangan turbin angin di wilayah pesisir, didapatkan bahwa dengan dimensi turbin angin dengan tinggi 1,427 m dan

dimeter 1,713 m dengan kecepatan angin 2,5 m/s dapat menghasilkan daya output sebesar 77,312W [1].

Sistem monitoring kecepatan angin pada turbin angin difungsikan untuk mendapatkan data kecepatan angin yang akurat, serta sebagai data acuan dalam menentukan lokasi pemasangan turbin angin. Anemometer merupakan piranti yang penting untuk mendapatkan data kecepatan angin, dengan pemasangan anemometer pada turbin angin dapat dimungkinkan monitoring lokasi pemasangan turbin angin secara kontinyu. Dalam pembuatan alat pengukur kecepatan angin dapat memanfaatkan mikrokontroler ATmega32 yang dapat diprogram sesuai dengan kebutuhan alat. Pada alat ukur kecepatan angin ini digunakan anemometer *cup* dengan modul sensor optocoupler, sebagai alat ukur untuk mendeteksi jumlah putaran *cup* anemometer. Sehingga pada tugas akhir ini dirancang sebuah sistem monitoring kecepatan angin menggunakan Visual Studio 2013 dengan menggunakan database MYSQL dan data logger MMC.

1.2 Rumusan Masalah

Pada pelaksanaan tugas akhir ini permasalahan yang diangkat adalah sebagai berikut:

- Bagaimana membuat alat pengukur kecepatan angin (*anemometer cup*) dengan menggunakan modul sensor *optocoupler* berbasis mikrokontroler ATmega32?
- Bagaimana membuat sistem perekam data *monitoring* kecepatan angin (*anemometer cup*) yang terintegrasi dengan *plant* turbin angin tipe horizontal axis?
- Bagaimana membuat sistem monitoring kecepatan angin (*anemometer cup*) menggunakan Visual Studio 2013 dengan menggunakan penyimpanan database MYSQL dan data logger MMC?

1.3 Batasan Masalah

Perlu diberikan beberapa batasan permasalahan agar pembahasan tidak meluas dan menyimpang dari tujuan. Adapun

batasan permasalahan dari sistem yang dirancang ini adalah sebagai berikut:

- Fokus dari tugas akhir ini adalah membahas tentang sistem monitoring kecepatan angin pada turbin angin, dan pada perancangan ini menggunakan mikrokontroler ATmega8535 sebagai fungsi pengolah data monitoring pada turbin angin tipe horizontal axis.
- Untuk *software* monitoring menggunakan Visual Studio 2013 dan untuk Database dalam menyimpan data Monitoring menggunakan MySQL.
- Menggunakan data logger MMC.
- Jenis Turbin yang digunakan yaitu turbin angin horizontal axis dengan tiga buah *blade*.

1.4 Tujuan

Tujuan yang dicapai dalam tugas akhir ini adalah sebagai berikut:

- Dapat membuat alat pengukur kecepatan angin (*anemometer cup*) dengan menggunakan modul sensor *optocoupler* berbasis mikrokontroler ATmega32.
- Dapat membuat sistem perekam data *monitoring* kecepatan angin (*anemometer cup*) yang terintegrasi dengan plant turbin angin tipe horizontal axis.
- Dapat membuat sistem monitoring kecepatan angin (*anemometer cup*) menggunakan Visual Studio 2013 dengan menggunakan penyimpanan database MySQL dan data logger MMC.

1.5 Metodologi Penelitian

Untuk mencapai tujuan dalam menyelesaikan tugas akhir yang direncanakan, maka perlu dilakukan suatu langkah-langkah. Adapun langkah – langkahnya adalah sebagai berikut :

- Studi Literatur, dalam merealisasikan tugas akhir ini tentu dibutuhkan referensi yang meliputi pengetahuan dasar tentang sistem monitoring kecepatan angin pada turbin angin.

- Mendesain ulang semua komponen – komponen dari sistem yang ada.
- Pembuatan sistem dari turbin angin:
 - Membuat mekanik dari perancangan *hardware*.
 - Membuat rangkaian pembentuk sistem keseluruhan dari perancangan penulis.
 - Membuat sistem monitoring dan kecepatan angin dengan Sistem Database dan Data logger.
 - Mengintegrasikan rangkaian dari alat ke mikrokontroler.
- Pengujian sistem dan analisa alat, dilakukan untuk mengetahui cara kerja dan hasil dari proses monitoring serta pengambilan data.
- Penyusunan Laporan, menyusun hasil teori dari pembuatan *hardware* dan *software*, analisa data dan kesimpulan dari data dan sistem yang ada.

1.6 Sistematika Laporan

Sistematika laporan yang digunakan dalam penyusunan laporan Tugas Akhir ini adalah sebagai berikut:

Bab I PENDAHULUAN

Berisi tentang latar belakang, permasalahan, batasan masalah, tujuan, metodologi dan sistematika laporan.

Bab II TEORI PENUNJANG

Berisi tentang landasan teori mengenai perancangan hardware dan software, *plant* turbin angin tipe horizontal axis, dan modul sensor *optocoupler* yang berkaitan dalam pembuatan Tugas Akhir sistem monitoring kecepatan angin.

Bab III PERANCANGAN DAN PEMBUATAN ALAT

Berisi tentang desain sistem monitoring kecepatan angin dan pembuatan database hasil monitoring yang meliputi sensor dan aplikasi.

Bab IV ANALISA DATA DAN PEMBAHASAN

Berisi tentang analisa hasil sistem monitoring kecepatan angin, pembuatan database serta data logger pada *Plant Wind Turbine* dan Pembahasannya.

Bab V PENUTUP

Berisi tentang hasil yang diperoleh dari analisis sistem, analisa data dan saran. Lampiran beserta daftar pustaka.

BAB II

TEORI PENUNJANG

2.1 Potensi Angin

Angin adalah udara yang bergerak yang diakibatkan oleh rotasi bumi dan juga karena adanya perbedaan tekanan udara disekitarnya. Angin bergerak dari tempat bertekanan udara tinggi ke bertekanan udara rendah. Apabila dipanaskan, udara memuai. Udara yang telah memuai menjadi lebih ringan sehingga naik. Apabila hal ini terjadi, tekanan udara turun karena udaranya berkurang. Penyebab timbulnya angin adalah matahari. Bumi menerima radiasi sinar matahari secara tidak merata. Dengan demikian, daerah khatulistiwa akan menerima energi radiasi matahari lebih banyak daripada di daerah kutub, atau dengan kata lain, udara di daerah khatulistiwa akan lebih tinggi dibandingkan dengan udara di daerah kutub. Pertukaran panas pada atmosfer akan terjadi secara konveksi. Berat jenis dan tekanan udara yang disinari cahaya matahari akan lebih kecil dibandingkan jika tidak disinari. Perbedaan berat jenis dan tekanan inilah yang akan menimbulkan adanya pergerakan udara. Pergerakan udara ini merupakan prinsip dari terjadinya angin. Secara ilmiah, pada abad ke-17, seorang fisikawan Itali, Evangelista Torricelli, mendeskripsikan bahwa angin dihasilkan karena adanya perbedaan suhu udara, dan juga perbedaan kepadatan (akibat perbedaan suhu udara), di antara dua daerah. Apabila kita asumsikan bahwa Bumi tidak berotasi, permukaan yang datar, dan udara yang lebih hangat terjadi pada daerah khatulistiwa dibandingkan pada kutub. Pada bulan Oktober – April, matahari berada pada belahan langit Selatan, sehingga benua Australia lebih banyak memperoleh pemanasan matahari dari benua Asia. Akibatnya di Australia terdapat pusat tekanan udara rendah (depresi) sedangkan di Asia terdapat pusat tekanan udara tinggi (kompresi). Keadaan ini menyebabkan arus angin dari benua Asia ke benua Australia. Di Indonesia angin ini merupakan angin musim Timur Laut di belahan bumi Utara dan angin musim Barat di belahan bumi Selatan. Oleh karena angin ini melewati Samudra

Pasifik dan Samudera Hindia maka banyak membawa uap air, sehingga pada umumnya di Indonesia terjadi musim penghujan. Musim penghujan meliputi seluruh wilayah Indonesia, hanya saja persebarannya tidak merata. Semakin ke timur curah hujan makin berkurang karena kandungan uap airnya semakin sedikit. Pada bulan April-Oktober, matahari berada di belahan langit utara, sehingga benua Asia lebih panas daripada benua Australia. Akibatnya, di Asia terdapat pusat tekanan udara rendah, sedangkan di Australia terdapat pusat tekanan udara tinggi yang menyebabkan terjadinya angin dari Australia menuju Asia. Di Indonesia terjadi angin musim timur di belahan bumi selatan dan angin musim barat daya di belahan bumi utara. Oleh karena tidak melewati lautan yang luas maka angin tidak banyak mengandung uap air oleh karena itu pada umumnya di Indonesia terjadi musim kemarau, kecuali pantai barat Sumatera, Sulawesi Tenggara, dan pantai selatan Irian Jaya. Antara kedua musim tersebut ada musim yang disebut musim pancaroba (peralihan), yaitu musim kemarau yang merupakan peralihan dari musim penghujan ke musim kemarau, dan musim labuh yang merupakan peralihan musim 5 kemarau ke musim penghujan [3]. Energi yang dibangkitkan oleh angin selama per jam dapat dinyatakan dengan rumus :

$$W = P \times t \dots\dots\dots (2.1)$$

Dimana :

W = energi (watt jam)

P = daya (watt)

t = waktu (detik)

untuk mengetahui daya atau energi yang dikeluarkan oleh alternator berdasarkan kecepatan angin dan diameter baling-baling dapat dinyatakan dengan rumus :

$$P = 1/12 \times V^3 \times D^2 \dots\dots\dots (2.2)$$

Dimana :

P = daya atau energi (watt)

V = kecepatan aliran udara (m/det)

D = diameter baling-baling (m)

2.2 Turbin Angin

Turbin angin adalah sebuah sistem yang berfungsi untuk mengubah energi kinetik angin menjadi energi mekanik pada poros turbin tersebut [9]. Energi angin dikonversi sebagian menjadi energi putar oleh rotor. Dengan atau tanpa roda gigi, putaran rotor tersebut biasanya digunakan untuk memutar generator yang akan menghasilkan energi listrik.

Untuk memproduksi listrik bolak-balik (ac) sistem ini harus didesain untuk selalu beroperasi pada kecepatan sudut yang tetap pada kecepatan angin yang berubah-ubah agar didapat frekuensi yang konstan. Dalam perencanaan kincir angin, adalah penting untuk menjaga perbandingan daya dan berat sekecil mungkin. Ini mengurangi tegangan yang diakibatkan oleh daya sentrifugal sudu. Secara teoritis jumlah daya yang bisa diserap oleh kincir angin dari angin adalah 59% untuk turbine dengan cerobong dan untuk turbin terbuka kira-kira 50-57% dari harga ini karena adanya kebocoran dan efek-efek lain. Daya yang dihasilkan oleh kincir angin secara langsung tergantung dari luas daerah yang disapu oleh sudu dan gaya angin per satuan luas yang tegak lurus pada kecepatan angin dibanding dengan kecepatan angin pangkat tiga [3].

2.3 *Horizontal Axis Wind Turbine*

Turbin angin sumbu horizontal merupakan turbin angin yang sumbu rotasi rotornya paralel terhadap permukaan tanah. Turbin angin sumbu horizontal memiliki poros rotor utama dan generator listrik di puncak menara dan diarahkan menuju dari arah datangnya angin untuk dapat memanfaatkan energi angin. Rotor turbin angin kecil diarahkan menuju dari arah datangnya angin dengan pengaturan baling – baling angin sederhana sedangkan turbin angin besar umumnya menggunakan sensor angin dan motor yang mengubah rotor turbin mengarah pada angin. Berdasarkan prinsip aerodinamis, rotor turbin angin sumbu horizontal mengalami gaya *lift* dan gaya *drag*, namun gaya *lift*

jauh lebih besar dari gaya *drag* sehingga rotor turbin ini lebih dikenal dengan rotor turbin tipe *lift*.

Dilihat dari jumlah sudunya, turbin angin sumbu horizontal terbagi menjadi sebagai berikut:

- Turbin angin satu sudu (*single blade*).
- Turbin angin dua sudu (*double blade*)
- Turbin angin tiga sudu (*three blade*)
- Turbin angin banyak sudu (*multi blade*)

2.4 Teori Turbin Angin

Dalam menentukan nilai daya keluaran pada perancangan wind turbine dibutuhkan beberapa teori dalam menentukan parameter-parameter. Adapun persamaan – persamaan tersebut adalah sebagai berikut:

a) *Chort*

Dalam menentukan nilai *chort* dibutuhkan parameter berupa *aspect ratio*. Pada perancangan *wind turbine* ini digunakan nilai *aspect ratio* 8, karena paling optimal.

$$\text{Aspect Ratio} = \frac{\text{Span}}{\text{Chort}} \dots \dots \dots (2.3)$$

Dimana : *Aspect Ratio* = 8

Span = Tinggi *blade* (m)

Chort = lebar *blade* (m)

b) Jari – Jari

Dalam menentukan nilai jari – jari *blade wind turbine* dibutuhkan parameter berupa *coefficient of power* (*Cp*). Pada perancangan *wind turbine* ini digunakan nilai *Cp* 4, karena paling optimal.

$$C_p = \frac{\text{Span}}{R} \dots \dots \dots (2.4)$$

Dimana : *Coefficient of power* (*Cp*) = 4

Span = Tinggi *blade* (m)

R = Jari – jari *blade* (m)

c) *Aswept*

Adapun persamaan dalam menentukan luasan sapuan dari *blade wind turbine* adalah sebagai berikut:

$$Aswept = \pi \times D \times S \dots\dots\dots(2.5)$$

Dimana : *Aswept* = luasan sapuan *blade* (m)

$$\pi = 3,14$$

D = diameter (m)

S = Tinggi *blade* (m)

d) Daya minimum (*P_w*)

Adapun persamaan dalam menentukan daya minimum *wind turbine* adalah sebagai berikut:

$$P_w = 0,625 \times Aswept \times V^3 \dots\dots\dots(2.6)$$

Dimana : *Aswept* = luasan sapuan *blade* (m)

V = laju angin (m/s)

e) Efisiensi *Wind Turbine*

Adapun persamaan dalam menentukan efisiensi *wind turbine*, dengan asumsi $\lambda = 1,6 \rightarrow 1,5 \leq \lambda \leq 2,5$ adalah sebagai berikut:

$$\eta_{wt} = 0,055 \lambda + 0,399 \dots\dots\dots(2.7)$$

f) *Turbine Rotational Speed*

Adapun persamaan dalam menentukan *Turbine Rotational Speed* adalah sebagai berikut:

$$\omega = \frac{\lambda \times v}{R} \times \frac{60}{2\pi} \dots\dots\dots(2.8)$$

Dimana :

λ = the tip speed ratio

v = the wind speed

R = radius wind turbine

g) *Torque Produced*

Adapun persamaan dalam menentukan *Torque Produced* adalah sebagai berikut:

$$T = \frac{P_w}{\omega} \dots\dots\dots (2.9)$$

Dimana :

P_w = power in the wind (Watt)

ω = turbine rotational speed

h) Check/Proof Calculations

Adapun persamaan dalam menentukan *Check/Proof Calculations* adalah sebagai berikut:

$$\lambda = \frac{\omega \times R \times 2\pi}{v \times 60} \dots\dots\dots (2.10)$$

i) Solidity

Adapun persamaan dalam menentukan *Solidity* adalah sebagai berikut:

$$S = \frac{N_B \times C}{D} \dots\dots\dots (2.11)$$

Dimana

N_b = Number of turbine blade

C = the chord length of the blade

D = diameters

2.5 Anemometer

Anemometer adalah sebuah perangkat yang digunakan untuk mengukur kecepatan angin, dan merupakan salah satu yang banyak dipakai dalam bidang Meterologi dan Geofisika atau stasiun prakiraan cuaca. Satuan meterologi dari kecepatan angin adalah Knots (Skala Beaufort) umumnya satuan yang digunakan adalah meter per detik (m/s). Sedangkan satuan meterologi dari arah angin adalah $0^\circ - 360^\circ$ [11] . Anemometer harus ditempatkan didaerah terbuka. Anemometer sendiri terdapat dua tipe secara umum. Tipe tersebut adalah sebagai berikut:

1. Anemometer dengan tiga atau empat mangkok

Sensor terdiri dari tiga atau empat buah mangkok yang dipasang pada jari – jari yang berpusat pada suatu sumbu vertikal atau semua mangkok tersebut terpasang pada poros vertikal. Seluruh mangkok menghadap ke satu arah melingkar sehingga

bila angin bertiup maka rotor berputar pada arah tetap. Kecepatan putar dari rotor tergantung kepada kecepatan tiupan angin. Melalui suatu sistem mekanik raga gigi, perputaran rotor mengatur sistem akumulasi angka penunjuk jarak tiupan angin. Anemometer tipe “cup counter” hanya dapat mengukur rata – rata kecepatan angin selama satu periode pengamatan. Dengan alat ini penambahan nilai yang dapat dibaca dari satu pengamatan ke pengamatan berikutnya, menyatakan akumulasi jarak tempuh angin selama waktu dari kedua pengamatan tersebut, sehingga kecepatan anginnya adalah sama dengan akumulasi jarak tempuh tersebut dibagi lama selang waktu pengamatannya.



Gambar 2.1. Anemometer Cup

2. Anemometer Termal

Anemometer merupakan satu sensor yang digunakan untuk mengukur kecepatan fluida (angin) sesaat. Cara kerja dari sensor ini berdasarkan pada jumlah panas yang hilang secara konvektif dari sensor ke lingkungan sekeliling sensor. Besarnya panas yang dipindahkan dari sensor secara langsung berhubungan dengan kecepatan fluida yang melewati sensor. Jika hanya kecepatan fluida yang berubah, maka panas yang hilang bisa diinterpretasikan sebagai kecepatan fluida tersebut. Kerja Anemometer mengikuti prinsip tabung pitot, yaitu dihitung dari tekanan statis dan tekanan kecepatan.

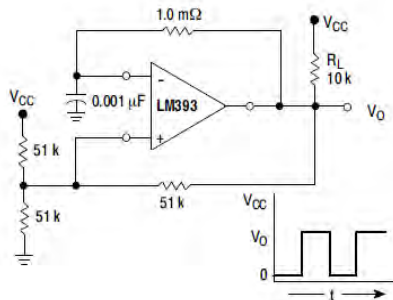
2.6 Sensor Optocoupler

Sensor *Optocoupler* merupakan alat yang digunakan untuk mendeteksi perubahan sinar inframerah, seperti yang terlihat pada Gambar 9.4. Sensor ini banyak dipakai untuk mendeteksi jarak ataupun pergerakan suatu benda dengan cara memberikan kisi-kisi ataupun baling-baling sehingga akan terdapat celah dan penghalang. Cara kerja dari sensor *optocoupler* adalah bila terhalang maka *output* akan open, dan bila tidak terhalang *output* akan *short*. Dengan cara kerja tersebut, sinar inframerah akan putus-putus dan menimbulkan pulsa-pulsa listrik. Pulsa pulsa itu kemudian dapat diolah dan ditangkap oleh mikrokontroler.



Gambar 2.2. Modul Sensor Optocoupler

Bagian dari sensor *optocoupler* ini adalah : sebuah led merah biasa atau led inframerah yang berfungsi sebagai *transmitter* dan sebuah fototransistor yang difungsikan sebagai *receiver*. Pada bagian *transmitter* dihubungkan pada tegangan yang difungsikan untuk menghidupkan led dan bagian *receiver* dihubungkan secara seri pada sumber tegangan dan lainnya menjadi terminal keluaran. Pada modul sensor tersebut digunakan rangkaian pengkondisian sinyal karena level tegangan keluaran dari sensor tersebut berbeda dengan level tegangan masukan mikrokontroler. Tegangan keluaran dari sensor berkisar antara 3Volt-5Volt sedangkan level tegangan mikrokontroler adalah 0Volt-5Volt. Untuk Rangkaian pengkondisian sinyal menggunakan komparator regeneratif (*schmitt trigger*) yang difungsikan sebagai pembalik dengan VR (variable resistor) yang disusun menggunakan IC LM393 seperti pada gambar 2.5.



Gambar 2.3. Rangkaian Pengkondisian Sinyal LM393

2.7 Mikrokontroler ATmega32

Mikrokontroler AVR (*Alf and Vegard's Risc Processor*) memiliki arsitektur 8 *bit*, dimana semua instruksi dalam kode 16 *bit* (16-bits word) dan sebagian besar instruksi dieksekusi dalam 1 siklus *clock*. Mikrokontroler AVR berteknologi RISC (*Reduced Instruction Set Computing*). Secara umum, AVR dikelompokkan menjadi 4 kelas, yaitu keluarga Attiny, keluarga AT90Sxx, keluarga ATmega dan keluarga AT86RFxx [13].

Mikrokontroler AVR ATmega32 memiliki arsitektur sebagai berikut:

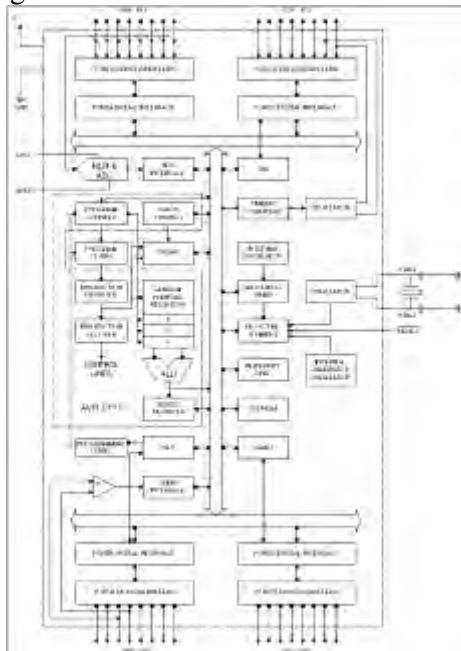
- a. Saluran I/O sebanyak 32 buah, yaitu Port A, Port B, Port C, dan Port D.
- b. ADC 10 *bit* sebanyak 8 saluran.
- c. Tiga buah *Timer/Counter* dengan kemampuan perbandingan.
- d. CPU yang terdiri atas 32 register.
- e. *Watchdog Timer* dengan osilator internal.
- f. SRAM sebesar 512 *byte*.
- g. Memori *flash* sebesar 8 KB dengan kemampuan *Read While Write*.
- h. Unit interupsi internal dan eksternal.
- i. Port antarmuka SPI.
- j. EEPROM sebesar 512 *byte* yang dapat diprogram saat operasi.

- k. Antarmuka komparator analog.
- l. Port USART untuk komunikasi serial.

Mikrokontroler AVR ATmega32 memiliki fitur sebagai berikut:

- a. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
- b. Kapabilitas memori *flash* 8 KB, SRAM sebesar 512 *byte* dan EEPROM sebesar 512 *byte*.
- c. ADC internal dengan 10 bit sebanyak 8 saluran.
- d. Portal komunikasi serial (USART) dengan kecepatan maksimal 2,5 Mbps.
- e. Enam pilihan mode *sleep* untuk menghemat penggunaan daya listrik.

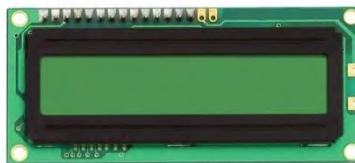
Blok diagram fungsional mikrokontroler ATmega32 ditunjukkan pada gambar 2.4.



Gambar 2.4. Blok diagram fungsional ATmega32

2.8 LCD (*Liquid Crystal Display*) 16x2

LCD (Liquid Cristal Display) berfungsi untuk menampilkan karakter angka, huruf atau simbol dengan baik dan konsumsi arus yang lebih rendah. *LCD (Liquid Cristal Display)* dot matrik terdiri dari bagian penampil karakter (*LCD*) yang berfungsi untuk menampilkan karakter dan bagian sistem prosesor *LCD* dalam bentuk rangkaian modul dengan mikrokontroler yang diletakan dibagian belakang *LCD* tersebut yang berfungsi untuk mengatur tampilan LCD serta mengatur komunikasi antara LCD dengan mikrokontroler. Modul prosesor pada LCD memiliki memori tersendiri sebagai berikut: *CGROM (Character Generator Read Only Memory)*, *CGRAM (Character Generator Random Access Memory)*, *DDRAM (Display Data Random Access Memory)*.



Gambar 2.5. Bentuk LCD (*Liquid Crystal Display*) 2×16

DB0 – DB7 adalah jalur data (*data bus*) sebagai jalur komunikasi dalam mengirimkan dan menerima data atau instruksi dari mikrokontroller ke modul LCD ini. RS yaitu pin yang berfungsi sebagai selektor register (*register sellect*) dimana dengan memberikan logika *low* (0) sebagai *register* perintah dan logika *high* (1) sebagai *register* data. R/W adalah pin dengan fungsi menentukan mode baca atau tulis dari data yang terdapat pada DB0 – DB7, yaitu dengan memberikan logika *low* (0) untuk fungsi *read* dan logika *high* (1) untuk mode *write*. Pin *Enable* (E), berfungsi sebagai *Enable Clock LCD*, berlogika 1 setiap kali pengiriman atau pembacaan data.

2.9 Karakteristik Alat Ukur

Setiap Instrumen ukur mempunyai karakteristik yang melekat padanya. Terdapat dua karakteristik instrument ukur yang digunakan, yaitu karakteristik statik dan karakteristik dinamik.

2.9.1 Karakteristik Statik Alat Ukur

Karakteristik statis instrumen merupakan hubungan antara output sebuah elemen (instrumen) dengan inputnya ketika inputnya konstan maupun berubah perlahan. Karakteristik statis tidak bergantung pada waktu. Yang termasuk dalam karakteristik statis adalah range, linieritas, sensitivitas, resolusi, akurasi, presisi, toleransi.

a. Range

Range adalah nilai minimum hingga maksimum suatu elemen. Range terdiri dari range input dan range output. Misalnya termocouple mempunyai range input 0°C - 250°C dan output range 5-20 mV.

b. Span

Span merupakan selisih nilai maksimum dengan nilai minimum. Span terdiri dari span input dan span output. Contoh, termocouple yang mempunyai range input 0°C - 250°C dan range output 5-20 mV span inputnya 250°C , span outputnya 15 mV.

c. Linieritas

Pengukuran dapat dikatakan ideal saat hubungan antara input pengukuran (nilai sesungguhnya) dengan output pengukuran (nilai yang ditunjukkan oleh alat) berbanding lurus. Linieritas merupakan hubungan nilai input dan output alat ukur ketika terletak pada garis lurus. Garis lurus ideal merupakan garis yang menghubungkan titik minimum input/output dengan titik maksimum input/output. Berikut merupakan persamaannya:

$$O_{ideal} = KI + a \dots \dots \dots (2.12)$$

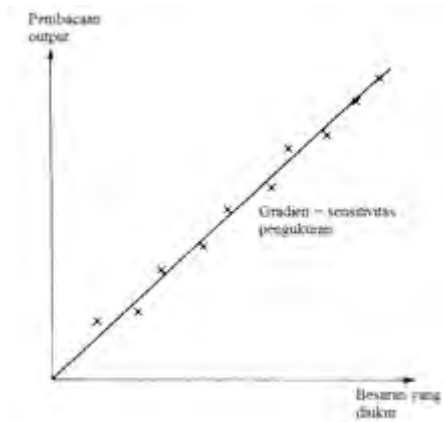
Dengan K adalah kemiringan garis, dapat diketahui melalui persamaan:

$$K = \frac{O_{max}-O_{min}}{I_{max}-I_{min}} \dots \dots \dots (2.13)$$

Dan a adalah pembuat nol (*zero bias*), dapat dihitung dengan persamaan:

$$a = O_{min} - KI_{min} \dots \dots \dots (2.14)$$

Berikut merupakan grafik karakteristik statis linier suatu instrumen:



Gambar 2.6 Karakteristik output instrumen linier

d. Non – Linieritas

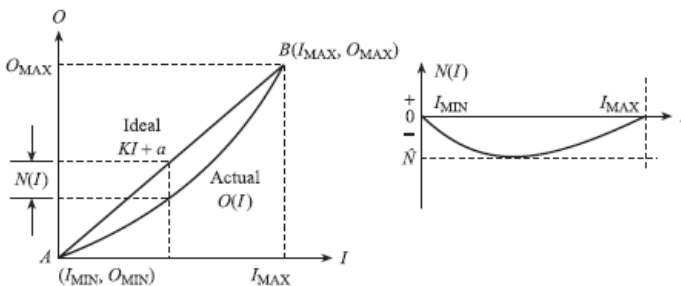
Dalam beberapa keadaan, bahwa dari persamaan linieritas muncul garis yang tidak lurus yang biasa disebut *non-linier* atau tidak linier. Didalam fungsi garis yang tidak linier ini menunjukkan perbedaan antara hasil pembacaan *actual* / nyata dengan garis lurus idealnya. Dengan persamaan sebagai berikut :

$$N(I) = O(I) - (KI + a) \dots\dots\dots(2.15)$$

$$O(I) = KI + a + N(I) \dots\dots\dots(2.16)$$

Sedangkan untuk persamaan dalam bentuk prosentase dari defleksi skala penuh, sebagai berikut:

Prosentase maksimum Nonlinieritas dari defleksi skala penuh = $\frac{\hat{N}}{O_{maks} - O_{min}} \times 100\% \dots\dots\dots(2.17)$



Gambar 2.7. Grafik non-linieritas (Bentley, 2005)

e. Sensitivitas

Sensitivitas diartikan seberapa peka sebuah sensor terhadap besaran yang diukur. Sensitivitas juga bisa diartikan sebagai perubahan output alat dibandingkan perubahan input dalam satu satuan. Sebagai contoh timbangan dengan kapasitas 700g mempunyai sensitivitas 1 mg. Ini berarti timbangan dapat digunakan untuk mengukur hingga 700g dengan perubahan terkecil yang dapat terbaca sebesar 1 mg.

$$\text{Sensitivitas} = \frac{\Delta O}{\Delta I} \dots \dots \dots (2.18)$$

f. Resolusi

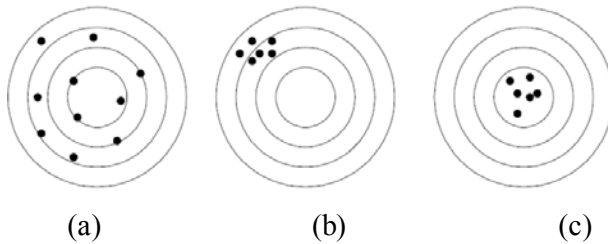
Resolusi merupakan perubahan terbesar dari input yang dapat terjadi tanpa adanya perubahan pada output. Suatu alat ukur dapat dikatakan mempunyai resolusi tinggi saat mampu mengukur dengan ketelitian yang lebih kecil. Misalkan, alat ukur yang mampu mengukur perubahan dalam mV mempunyai resolusi yang lebih tinggi dibanding alat ukur yang mengukur perubahan dalam skala volt.

g. Akurasi

Akurasi merupakan ketepatan alat ukur untuk memberikan nilai pengukuran yang mendekati nilai sebenarnya. Karena pada eksperimen nilai sebenarnya tidak pernah diketahui oleh sebab itu diganti dengan suatu nilai standar yang diakui secara konvensional. Contohnya termometer dengan skala 0°-150°C mempunyai akurasi 1°C, ini berarti jika termometer menunjukkan nilai 80°C maka nilai sebenarnya adalah 79°C-81°C.

h. Presisi

Presisi adalah kemampuan instrument/elemen untuk menampilkan nilai yang sama pada pengukuran berulang singkat.



Gambar 2.8 Penjelasan Akurasi dan Presisi (a) akurasi rendah, presisi rendah (b) akurasi rendah, presisi tinggi (c) akurasi tinggi presisi tinggi

2.10 Ketidakpastian Kalibrasi

Ketidakpastian adalah nilai ukur sebaran kelayakan yang dapat dihubungkan dengan nilai terukurnya. Dimana di dalam nilai sebaran tersebut terdapat nilai rentang yang menunjukkan nilai sebenarnya.

- Klasifikasi ketidakpastian, antara lain :
 - **Tipe A** : nilai ketidakpastian yang dilihat dari analisis pengukuran statistik (*ISO GUM B 2.15; VIM 3.5* dalam KAN DP.01.23).

Di dalam tipe ini dilakukan pengukuran hingga n kali, dimana dari pengukuran tersebut akan mendapatkan nilai rata-rata, standar deviasi, dan data keterulangan. Dimana rumus umum ketidakpastian tipe A sebagai berikut :

$$U_{a1} = \frac{\sigma}{\sqrt{n}} \dots\dots\dots (2.19)$$

Dimana :

σ = *Standart deviasi koreksi*

n = Jumlah data

$$\sigma = \frac{\sqrt{\sum (y_i - \bar{y})^2}}{n-1} \dots\dots\dots (2.20)$$

$$U_{a2} = \sqrt{\frac{SSR}{n-2}} \text{ (Ketidakpastian regresi)} \dots\dots\dots (2.21)$$

Dimana :

SSR (*Sum Square Residual*) = \sum SR (*Square Residual*)

SR = R^2 (*Residu*)

$$Y_i (\text{Nilai koreksi}) = t_i - x_i \dots \dots \dots (2.22)$$

$$Y_{reg} = a - (b \times t_i) \dots \dots \dots (2.23)$$

$$a = \bar{y}_i - (b \times \bar{t}_i) \dots \dots \dots (2.24)$$

$$b = \frac{n \cdot \sum t_i y_i - \sum y \cdot \sum t_i}{n \cdot \sum t_i^2 - (\sum t_i)^2}; \dots \dots \dots (2.25)$$

Dimana :

t_i = Pembacaan standar

x_i = Pembacaan alat

y_i = Nilai koreksi

- **Tipe B** : nilai ketidakpastian yang tidak dilihat dari analisis pengukuran statistik (*ISO GUM B 2.15; VIM 3.5* dalam KAN DP.01.23). Berikut merupakan rumus umum dari ketidakpastian tipe B :

$$U_{B1} = \frac{\frac{1}{2} \times \text{Resolusi}}{\sqrt{3}} \dots \dots \dots (2.26)$$

$$U_{B2} = \frac{a}{k} \dots \dots \dots (2.27)$$

Dimana :

U_{B1} = Nilai ketidakpastian resolusi

U_{B2} = Nilai ketidakpastian dari alat standar/kalibrator

- **Ketidakpastian Baku Gabungan (Kombinasi)**
Ketidakpastian baku gabungan disimbolkan dengan U_c , dimana nilai ketidakpastian yang digunakan untuk mewakili nilai estimasi standar deviasi dari hasil pengukuran. Nilai ketidakpastian baku gabungan didapat dari menggabungkan nilai-nilai ketidakpastian

baku dari setiap taksiran masukan (hukum propagasi ketidakpastian) (*ISO GUM B 2.15*; *VIM 3.5* dalam KAN DP.01.23).

Berikut merupakan rumus umum ketidakpastian baku gabungan :

$$U_c = \sqrt{U_{A1}^2 + U_{A2}^2 + U_{B1}^2 + U_{B2}^2} \dots\dots\dots(2.28)$$

Dimana :

U_c = Nilai ketidakpastian kombinasi

U_{A1} = Nilai ketidakpastian hasil pengukuran

U_{A2} = Nilai ketidakpastian regresi

U_{B1} = Nilai ketidakpastian resolusi

U_{B2} = Nilai ketidakpastian kalibrator

- **Derajat Kebebasan Efektif**

Derajat kebebasan efektif ini berfungsi sebagai pemilihan faktor pengali untuk distribusi *Student's T* serta sebagai penunjuk perkiraan kehandalan ketidakpastian (*ISO GUM B 2.15*; *VIM 3.5* dalam KAN DP.01.23). Derajat kebebasan disimbolkan dengan v , dengan rumus sebagai berikut :

$$V = n-1 \dots\dots\dots(2.29)$$

Dimana :

n = Jumlah data

Sedangkan untuk derajat kebebasan efektif merupakan estimasi dari derajat kebebasan ketidakpastian baku gabungan yang dirumuskan sebagai berikut (rumus *Welch-Setterthwaite*):

$$V_{eff} = \frac{(U_c)^4}{\sum (U_i)^4 / v_i} \dots\dots\dots(2.30)$$

Dimana :

V_{eff} = Derajat kebebasan efektif dari ketidakpastian kombinasi

v_i = Derajat kebebasan dari komponen ketidakpastian ke-i

U_i = Hasil ketidakpastian tipe A dan B

Setelah ditentukan nilai derajat kebebasan efektif, maka dapat dilanjutkan dengan menghitung nilai faktor cakupan sesuai dengan tingkat kepercayaan yang diinginkan, dimana faktor cakupan (k) didapat dari tabel *T-students*.

- **Ketidakpastian Diperluas, U_{exp}**

Ketidakpastian diperluas merupakan akhir nilai ketidakpastian dengan tingkat kepercayaan. Tingkat kepercayaan tingkat keyakinan mengenai daerah nilai sebenarnya pada suatu pengukuran (LPF, 2013).

$$U_{exp} = k \times U_c \dots\dots\dots (2.31)$$

Dimana :

k = Faktor cakupan

U_c = Nilai ketidakpastian kombinasi

2.11 Microsoft Visual Studio 2013

Microsoft Visual Studio selain disebut sebagai sebuah bahasa pemrograman, juga disebut sebagai sarana (tool) untuk menghasilkan program-program aplikasi berbasis Windows. Beberapa kemampuan atau manfaat dari Visual Studio di antaranya seperti :

1. Untuk membuat program aplikasi berbasis Windows.
2. Untuk membuat objek-objek pembantu program seperti misalnya kontrol ActiveX, file help, aplikasi internet, dan sebagainya.
3. Menguji program (debugging) dan menghasilkan program akhir berakhiran EXE yang bersifat executable atau dapat langsung dijalankan.

Visual Basic adalah perangkat lunak yang berjalan di atas platform sistem operasi Microsoft Windows. Didalam program Visual Basic ini akan digunakan sebagai tampilan pada

monitoring kecepatan, arus, tegangan pada *Miniplant* PLTM dengan membuat program dari data yang keluar dari sensor dibaca dalam mikrokontroller kemudian dapat diintegrasikan dalam program ini dan tersimpan dalam database sehingga semuanya terintegrasi dengan baik.

2.12 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.

2.13 MMC (*Multi Media Card*)

MMC (*Multi Media Card*) merupakan alat untuk menyimpan data digital. *Memory card* biasanya mempunyai kapasitas ukuran berdasarkan bit digital, yaitu 16 MB, 32 MB, dan seterusnya. Adapun konfigurasi dari pin MMC adalah sebagai berikut:



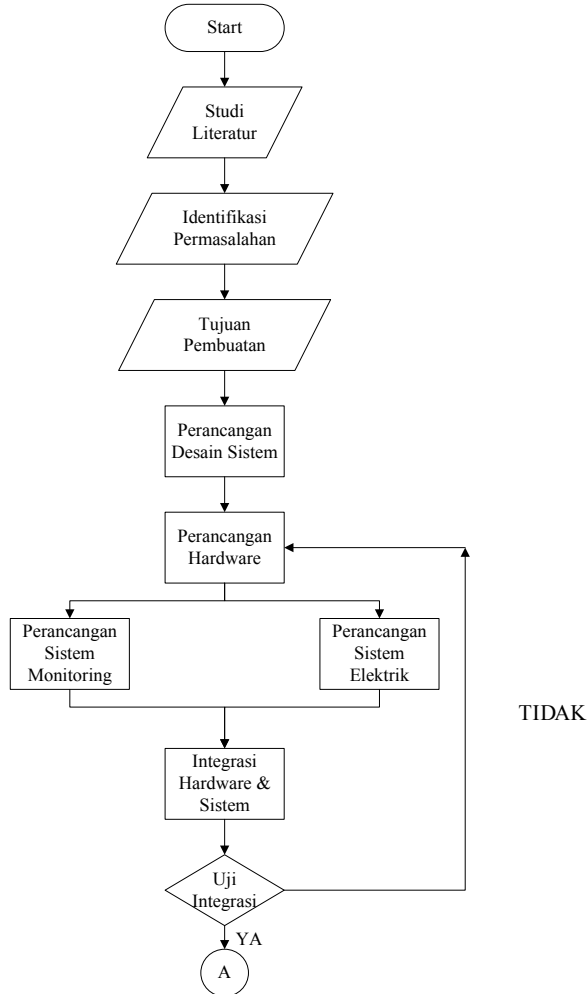
2.9 Konfigurasi MMC

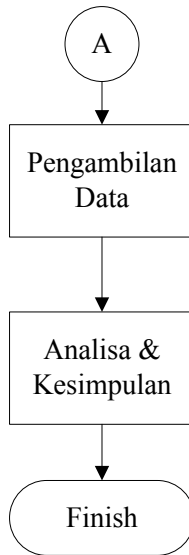
- Pin 1 = CS = *chip select*
- Pin 2 = DI = *data input*
- Pin 3 = Vss = *ground*
- Pin 4 = Vcc, tegangan 3,3 Volt
- Pin 5 = SCLK = *serial clock*
- Pin 6 = Vss2 = *ground*
- Pin 7 = D0 = *data out*
- Pin 8 = DAT1
- Pin 9 = DAT2

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1. Diagram Alir (*Flowchart*) Metodologi Penelitian

Tahapan penelitian Tugas Akhir ini, secara umum dapat digambarkan dalam *flowchart* seperti dibawah ini.





Gambar 3.1 *Flowchart* Tugas Akhir

3.2. Keterangan Flowchart

3.2.1 Studi Literatur

Mempelajari literatur dari penelitian – penelitian yang pernah dilakukan sebelumnya mengenai pembuatan alat ukur kecepatan angin serta proses monitoring dan akuisisi data. Selain belajar sistem pada perancangan alat ukur, juga mencari literatur mengenai pembuatan *plant wind turbine*, serta literatur yang berkaitan dengan elemen-elemen yang digunakan dalam sistem monitoring kecepatan angin, misalnya *datasheet* sensor dan karakteristik mikrokontroler.

3.2.2 Identifikasi Permasalahan

Melakukan identifikasi permasalahan yang berkaitan dengan pengukuran dan sistem *monitoring* kecepatan angin, misalnya konstruksi anemometer dan perancangan sistem monitoring kecepatan angin dengan visual studio 2013, MySQL, dan *data logger* dengan MMC. Selain itu identifikasi

permasalahan terdapat pada perancangan *plant wind turbine*, misalnya pada perancangan *blade*, dan keluaran daya minimum *wind turbine* terhadap kecepatan angin.

3.2.3 Tujuan Pembuatan

Pembuatan sistem monitoring kecepatan angin bertujuan untuk mengamati perbandingan daya keluaran dari *plant wind turbine* terhadap kecepatan angin. Selain itu monitoring kecepatan angin dapat menjadi alat bantu dalam memantau kecepatan angin, guna menjaga batas aman (*safety*) pada *plant wind turbine*.

3.2.4 Perancangan Blok Sistem

Pembuatan alat ukur dan monitoring kecepatan angin difungsikan untuk mengamati perubahan dari kecepatan angin, alat ukur ditempatkan pada sisi atas dari *wind turbine*. Untuk monitoring digunakan komunikasi serial dengan menggunakan UART (*Universal Asynchronous Receiver/Transmitter*) pada mikrokontroller ATmega32 dengan *software monitoring visual studio 2013*, *software database MYSQL*, dan *data logger* dengan MMC.

3.2.5 Perancangan Wind Turbine

Dengan mengacu pada landasan teori, maka didapat perhitungan untuk menentukan dimensi dari perancangan *wind turbine*.



Gambar 3.2 Konstruksi turbin angin horizontal axis

Tabel 3.1. Parameter *Wind Turbine*

No	Parameter	Nilai
1	<i>Aspect Ratio (AR)</i>	8
2	Span	35 cm
3	<i>Velocity</i>	3 m/s
4	Cp	2

• Kalkulasi Performansi

- a. Menentukan nilai *chort* :

$$Chort = 35/8 = 4,375 \text{ cm}$$

- b. Menentukan nilai Jari-Jari :

$$R = 35/2 = 17,5 \text{ cm}$$

- c. Menentukan luasan sapuan *blade* :

$$Aswept = 3,14 \times 70 \times 35 = 7693 \text{ cm}^2 = 0,7693 \text{ m}^2$$

- d. Menentukan daya minimal *available on wind turbine*

$$Pw = 0,625 \times 0,7693 \times 3^3 = 12,981$$

- Efisiensi *Wind Turbine*

$$\eta_{wt} = 0,055 \times 1,6 + 0,399 = 0,487 = 48,7 \%$$

- *Turbine Rotational Speed*

$$\omega = \frac{1,6 \times 3}{0,35} \times \frac{60}{2 \times 3,14} = 13,714 \times 9,55 = 131 \text{ rpm}$$

- *Torque Produced*

$$T = \frac{12,981}{131} = 0,099 \text{ N.m}$$

- *Check/Proof Calculations*

$$\lambda = \frac{131 \times 0,35 \times 2 \times 3,14}{3 \times 60} = 1,599 \cong 1,6$$

- *Solidity*

Untuk 3 kincir angin

$$S = \frac{3 \times 0,04375}{0,7} = 0,1875$$

3.2.6 Perancangan *Hardware*

Perancangan *hardware* terdiri dari perancangan sistem elektrik dan perancangan sistem *monitoring*. Dalam perancangan sistem elektrik terdapat beberapa tahap, yaitu pemilihan jenis modul *sensor optocoupler* sebagai sensor kecepatan, kemudian masuk ke pemrograman mikrokontroler ATmega32 untuk pembacaan keluaran dari sensor. Pada perancangan sistem *monitoring* kecepatan angin terdapat diagram blok pengukuran. Berikut merupakan diagram blok sistem *monitoring* kecepatan angin.

Diagram blok tersebut menjelaskan mengenai alur dari sistem *monitoring* kecepatan angin yang menggunakan modul *sensor optocoupler* sebagai sensor kecepatan angin. Mulai dari input oleh angin yang mendorong *cup anemometer*, modul sensor *optocoupler (sensing element)*, ATmega32 (*signal conditioning element and signal processing element*), dan pada *data presentation element* digunakan LCD dan *Visual Studio 2013*, serta penyimpanan data menggunakan MySQL dan MMC.

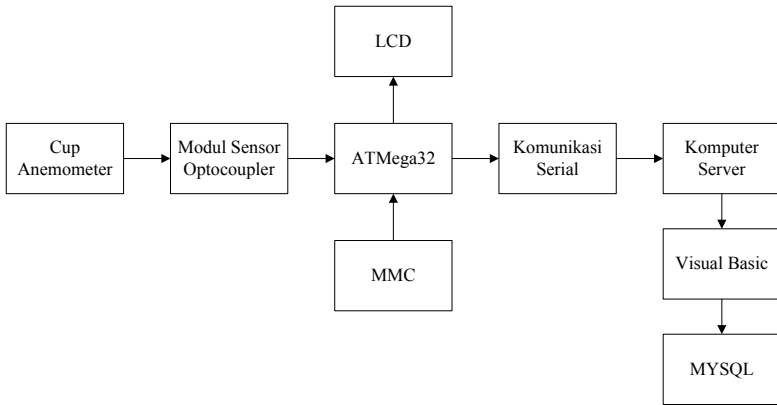
a. Perancangan dan Pembuatan Alat (*Hardware*)

Alat ukur kecepatan angin menggunakan modul *sensor optocoupler*, dengan menggunakan ic LM393 sebagai

komparator. Pengukuran kecepatan angin dilakukan oleh modul *sensor optocoupler* dengan mendeteksi sinyal gelap terang yang dihasilkan dari putaran *disk rotary encoder*, yang ditempatkan pada poros dari anemometer tipe *cup*. Keluaran dari sensor tersebut adalah berupa sinyal PWM (*Pulse Width Modulation*). Sinyal PWM tersebut selanjutnya akan di proses oleh mikrokontroler ATmega32 pada PIND.2 dengan memprogram eksternal *interrupt* dan *timer interrupt*. Eksternal *interrupt* pada ATmega32 difungsikan sebagai interupsi sistem untuk membaca output-an sensor yakni 0 dan 1. Pada *timer interrupt* difungsikan sebagai interupsi sistem untuk membaca output-an sensor yakni 0 dan 1, yang kemudian dicacah dan dimasukkan pada rumus kecepatan angin. Berikut ini merupakan rumus untuk mendapatkan timer 1 detik dalam pembacaan frekuensi sensor:

$$\begin{aligned} \text{Ttimer1} &= 0,00000025 \times (65536 - \text{TCNT1}) \times 1024 \\ 1 &= 0,00000025 \times (67108864 - 1024 \times \text{TCNT1}) \\ \text{TCNT1} &= 61629,75 \rightarrow \text{F0BD} \end{aligned}$$

Dari perhitungan tersebut maka akan didapatkan pencacahan frekuensi selama 1 detik. Hasil dari pembacaan sensor tersebut selanjutnya akan ditampilkan data pada LCD (*Liquid Crystal Display*) serta *interfacing* pada *Visual Studio 2013* dan *database MySQL*, serta data logger MMC.



Gambar 3.3 Diagram Blok Konstruksi Sensor Kecepatan Angin



Gambar 3.4 Rangkaian Modul *Sensor Optocoupler*



Gambar 3.5 Konstruksi *sensor optocoupler* dengan *disk rotary encoder*

Setelah *sensing element* dan *signal conditioning*, tahap selanjutnya adalah masuk ke *signal processing*. *Signal processing* dilakukan oleh mikrokontroler ATmega32. *Signal processing* bekerja dengan koding program AVR dengan menggunakan Eksternal *Interrupt* dan *Timer Interrupt*. Eksternal *interrupt* digunakan untuk mencacah pulsa 0 dan 1 pada pembacaan sensor terhadap putaran *disk rotary encoder*, sedangkan *Timer Interrupt* digunakan untuk penghitung kecepatan angin. Perhitungan dilakukan dengan cara perkalian antara pulsa yang disensing oleh modul *sensor optocoupler* dengan waktu per menit (60 detik) serta jumlah lubang *disk rotary encoder* (22 lubang), dari data ini didapat nilai RPM. Dari nilai RPM ini selanjutnya akan dikonversikan menjadi kecepatan angin, dikalikan dengan keliling baling baling per menit (60 detik) dan dikali 10 sebagai penyesuaian dengan alat ukur standar. Hasil data dari pemrosesan sinyal akan ditampilkan oleh LCD (*Liquid Crystal Display*) seperti pada gambar dibawah ini.

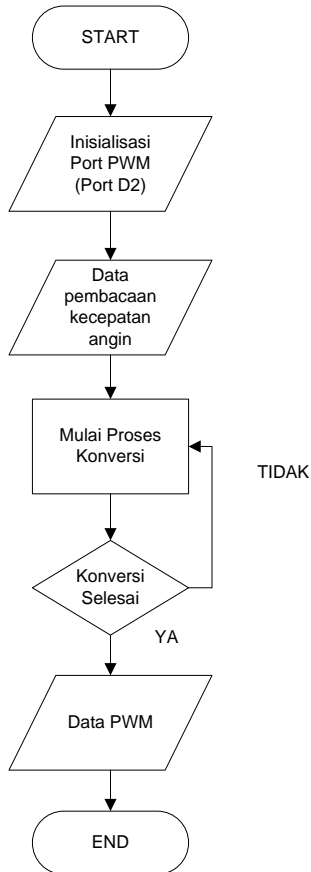


Gambar 3.6 *Display Hardware* alat ukur kecepatan angin

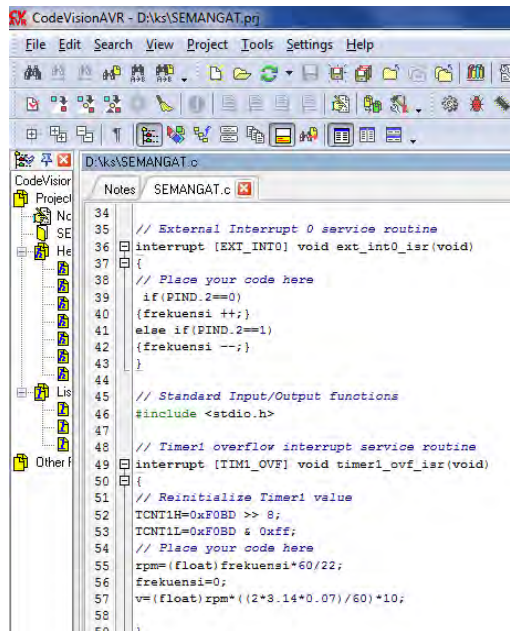
b. Perancangan dan Pembuatan *Software*

Perancangan *software* pada tahap ini merupakan siklus gabungan untuk rangkaian pemrosesan sinyal. Rangkaian pemrosesan sinyal pada alat ini menggunakan mikrokontroler ATmega32. Untuk membuat program pada mikrokontroler ATmega32, dibutuhkan *software* CVAVR. Pada program ini, akan diberikan koding untuk mengolah sinyal masukan. Sinyal masukan dari *sensing element*, sudah merupakan sinyal PWM.

Jadi, pada *programming* CVAVR digunakan fitur eksternal *interrupt* dan *timer interrupt* yang menggunakan PIND2.



Gambar 3.7 *Flowchart* Perancangan Sensor



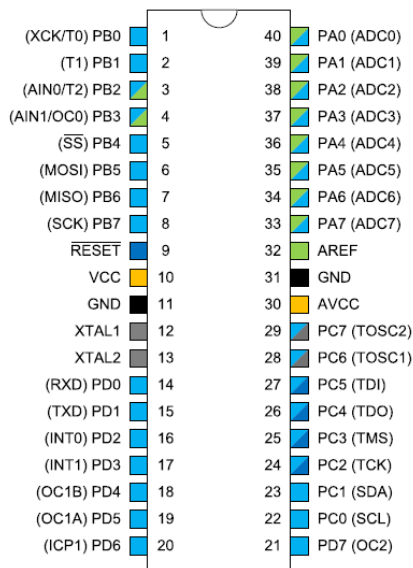
Gambar 3.8 *Software CVAVR*

3.2.7 Minimum System ATmega32

Konfigurasi pin dari mikrokontroler ATmega32 sebanyak 40 pin dapat dilihat pada Gambar 2.7. Dari gambar tersebut dapat dijelaskan secara fungsional konfigurasi pin Atmega32 sebagai berikut:

1. VCC merupakan pin yang berfungsi sebagai pin masukan catu daya
2. GND merupakan pin *ground*.
3. Port A (PA0...PA7) merupakan pin I/O dua arah dan pin masukan ADC.
4. Port B (PB0...PB7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu *Timer/Counter*, komparator analog dan SPI.

5. Port C (PC0...PC7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu TWI, komparator analog dan *Timer Oscillator*.
6. Port D (PD0...PD7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu komparator analog, interupsi eksternal dan komunikasi serial.
7. RESET merupakan pin yang digunakan untuk me-reset mikrokontroler.
8. XTAL1 dan XTAL2 merupakan pin masukan *clock* eksternal.
9. AVCC merupakan pin masukan tegangan untuk ADC.
10. AREF merupakan pin masukan tegangan referensi ADC.



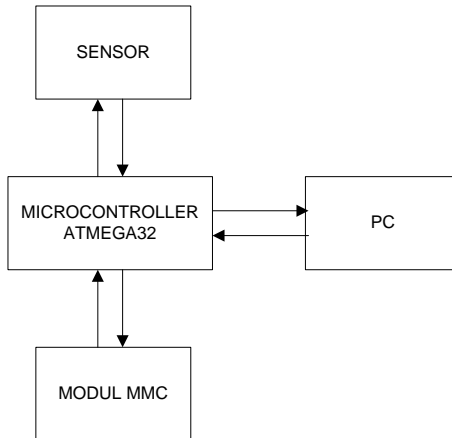
Gambar 3.9 Konfigurasi pin ATMega32

3.2.8 Integrasi *Hardware* dan *Software* Pemrograman

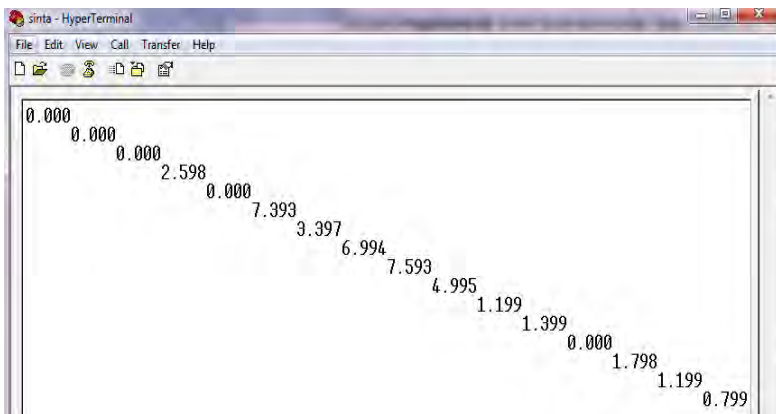
Ketika *coding* program sudah jadi maka dilakukan pengintegrasian antara *hardware* dengan *software*. Didalam *software* dilakukan penyamaan *Port Personal Computer (PC)*

yang digunakan untuk melakukan proses *uploading* program yang ada ke *microcontroller* ATmega32.

Penyesuaian *interface* antara *software* CVAVR dan *hardware* ATmega32 dapat dikonfigurasi melalui aplikasi *Hyperterminal*. Jika sudah berhasil, maka hasil program dapat dilihat pada LCD ataupun dari *serial monitor Hyperterminal*.



Gambar 3.10 Blok Integrasi Alat



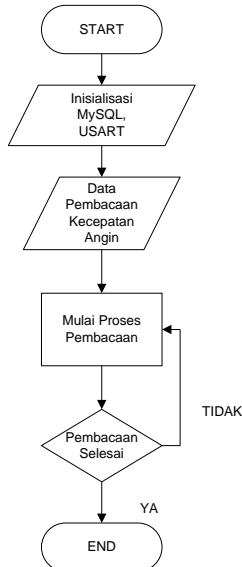
Gambar 3.11 *Display Hyperterminal* Komunikasi Serial alat ukur kecepatan angin

3.2.9 Pengujian Sistem Pengukuran

Pengujian sistem pengukuran ini dilakukan untuk mengetahui sudah berjalankah atau belum dari hasil *uploading* program dari *software* CVAVR ke *hardware* ATmega32. Jika dari *sensing element* masih belum dapat menampilkan data, maka proses pemrograman dan integrasi *software* serta *hardware* perlu diulang.

3.2.10 Pembuatan Desain HMI (*Human Machine Interface*) di PC dengan *Software Visual Studio 2013* serta media penyimpanan data

Setelah dilakukan proses integrasi *software* dan *hardware*, maka dilakukan pembuatan HMI pada PC (*Personal Computer*) serta penyimpanan data secara *real time*. Pembuatan tampilan ini menggunakan *software visual studio 2013*. Data yang akan ditampilkan secara *real time* adalah data pengukuran kecepatan angin. Desain yang telah dibuat dapat dilihat pada gambar dibawah ini.



Gambar 3.12 *Flowchart Perancangan Software Visual Studio*



Gambar 3.13 Tampilan Awal *Software* monitoring dengan *Visual Studio*

3.2.11 Integrasi *Hardware* Sistem *Monitoring* dan akuisisi data

Setelah dibuat *listing program* di *Visual Studio* kemudian dilakukan penghubungan antara program *Visual Basic* 2013 dengan *hardware* sistem *monitoring* yang telah dibuat sebelumnya.

BAB IV

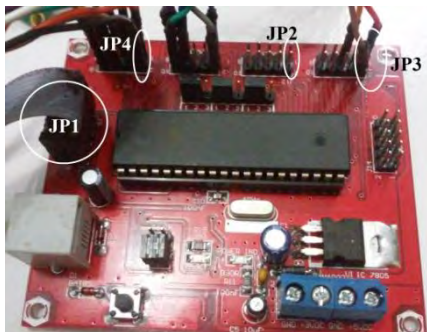
ANALISA DATA DAN PEMBAHASAN

4.1 Pengujian Alat

Setelah dilakukan perancangan dan pembuatan monitoring kecepatan angin pada *Miniplant Wind Turbine Horizontal Axis*, maka perlu dilakukan pengujian terhadap alat yang sudah dibuat serta melakukan analisa secara menyeluruh terhadap hasil pengujian tersebut. Hal ini dilakukan dengan tujuan untuk mengetahui performansi alat secara keseluruhan. Pengujian alat secara keseluruhan dilakukan pada perangkat keras (*hardware*) dan perangkat lunak (*software*). Untuk perangkat keras menggunakan modul sensor optocoupler sebagai media pembaca variabel kecepatan angin dengan tambahan *disk rotary encoder* 22 lubang. Dan juga mikrokontroler ATmega32 sebagai pemroses sinyal keluaran dari sensor dan dapat ditampilkan pembacaan sensor dengan tampilan dalam *Display* berupa LCD, software monitoring Visual Studio 2013, data logger MySQL, dan *data logger* dengan MMC.

4.1.1 Pengujian Tegangan *Minimum System* ATmega32

Setelah dilakukan uji coba pada mikrokontroler terhadap pengaruh tegangan output pada LCD, dan sensor didapat data sebagai berikut ini:



Gambar 4.1 *Minimum System* ATmega32

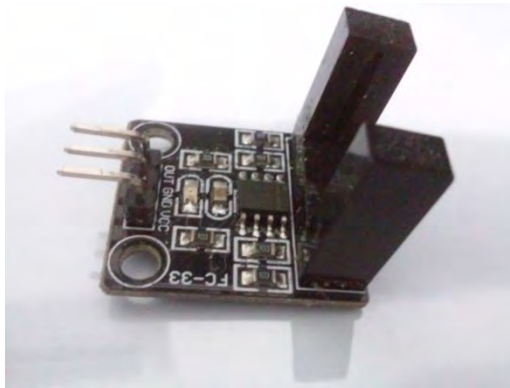
Tabel 4.1 Uji Tegangan Pada Mikrokontroler ATmega32

Port (JP)	Nilai Tegangan (Volt)	Keterangan
JP1	5,04	Sumber
JP2	4,92	VCC dan Ground Mikrokontroler
JP3	4,93	Sebelum di putar (VCC dan Ground Mikrokontroler Sensor)
	4,9	Sesudah di putar (VCC dan Ground Mikrokontroler Sensor)
JP4	4,89	VCC dan Ground Mikrokontroler LCD

4.1.2 Modul Sensor Optocoupler

Dalam pengujian modul sensor optocoupler, masukan sensor optocoupler adalah frekuensi dari setiap putaran disk rotary encoder terhadap photodiode yang terdapat pada modul sensor optocoupler. Setelah dapat diberikan sinyal tersebut maka modul sensor optocoupler dapat memberikan sinyal PWM kepada ATmega32 untuk diproses lebih lanjut menjadi nilai kecepatan angin, kemudian untuk blok diagram modul sensor optocoupler dapat dilihat pada Gambar 4.1. Untuk konfigurasi yang digunakan bisa dilihat pada Gambar 4.2.

**Gambar 4.2.** Diagram Blok Modul Sensor Optocoupler

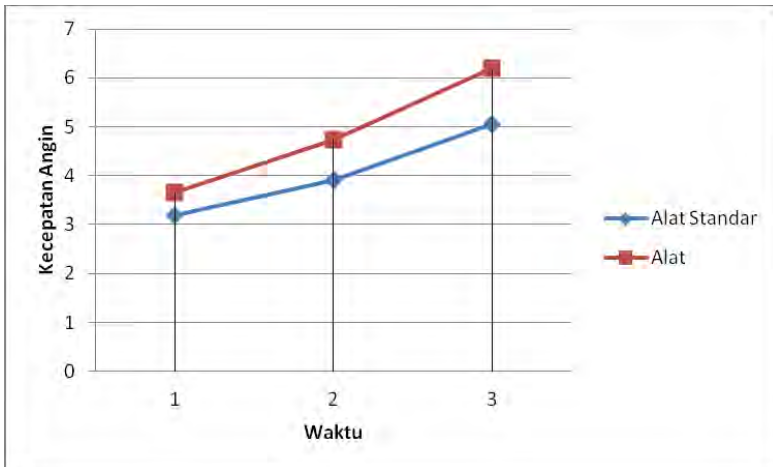


Gambar 4.3. Konfigurasi Modul Sensor Optocoupler

Setelah dilakukan pengujian sensor didapatkan data kalibrasi dengan membandingkan pembacaan sensor dengan pembacaan alat ukur standar. Pada tabel 4.1 dan gambar 4.3 dilampirkan hasil dari uji coba sensor terhadap alat ukur standar.

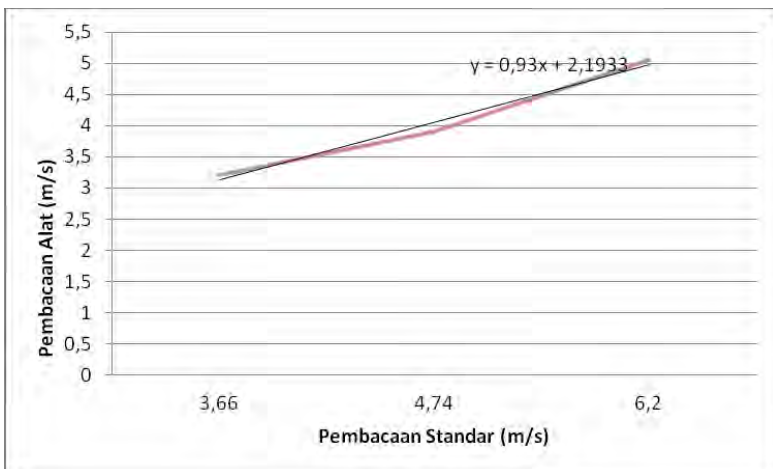
Tabel 4.2. Data Pengujian Alat

RENTANG	PEMB STD (m/s)	PEMBACAAN ALAT (m/s)										RAT A- RAT A PEM BAC AAN	ERRO R (m/s)
		PEMBACAAN NAIK					PEMBACAAN TURUN						
		1	2	3	4	5	1	2	3	4	5		
3,2	3,2	3,2	5,2	3,6	3	4	4	3,2	3,2	4	3,2	3,66	-0,4600
3,9	3,9	4,2	4,6	6	4	5	4,2	4,6	4,6	5,2	5	4,74	-0,8400
5,06	5,06	6,6	6,2	6,4	5,4	5,8	6,4	7,2	5,2	6	6,8	6,2	-1,1400



Gambar 4.4. Grafik Pembacaan Alat dan Pembacaan Standar

Jika tipe grafik dirubah menjadi tipe scatter, maka akan didapatkan grafik seperti pada Gambar 4.4.



Gambar 4.5. Grafik Perbandingan Uji dengan Standard Anemometer

Dari grafik pada Gambar 4.3 menunjukkan bahwa persamaan pengujian dari alat yang dibandingkan dengan alat ukur yang standar, dimana dari persamaan matematik yaitu $y = 0,93x + 2,1933$ menghasilkan data pembacaan alat ukur kecepatan angin yang sudah mendekati dengan pembacaan alat standar, anemometer.

Kemudian dilakukan perhitungan untuk menentukan karakteristik statik alat. Berdasarkan data yang telah didapatkan dari pengujian spesifikasi alat, menghasilkan data sebagai berikut:

Tabel 4.3. Data Karakteristik Statik Alat

NO	VOLT	PEMB. STD	RATA-RATA PENGUKURAN			BEDA HISTERISIS	(STD-ALAT)/STD	LINEARITAS (KI+a)	NON-LINEARITAS (output-linearitas)
			NAIK	TURUN	RATA-RATA PEMB (X)				
1	8	3,2	3,800	3,520	3,660	-0,280	0,14375	3,8	0,000
2	10	3,9	4,760	4,720	4,740	-0,040	0,215384615	4,658064516	0,102
3	12	5,06	6,080	6,320	6,200	0,240	0,225296443	6,08	0,000
	JUMLAH	12,16			15		0,584431058		0,102
	RATA-RATA	4,05			5		0,194810353		

Sehingga menghasilkan nilai:

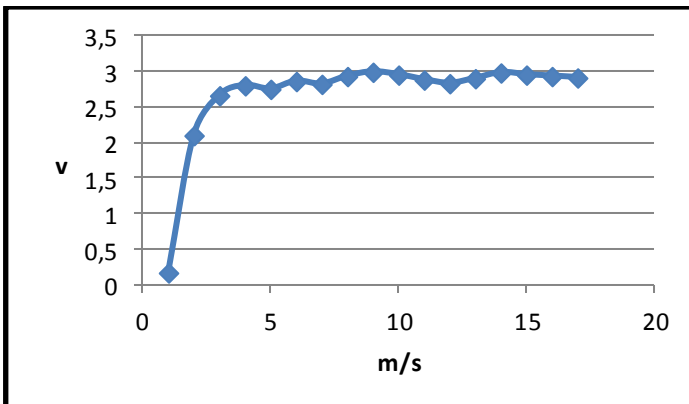
- Range* : 8-12 Volt/ 3,2 – 5,06 m/s
- Span* : 4 Volt/ 1,86 m/s
- Resolusi* : 0,01
- Non-linearitas* : 4,471%
- Histerisis* : 3,947
- Akurasi* : 79,934%
- Kesalahan* : 20,066%
- Sensitivitas* : 1,225806452

Tabel 4.4. Data Hubungan Antara Frekuensi ke Tegangan

kecepatan Angin m/s	Tegangan v
1	0,17
2	2,1

Tabel 4.4 Lanjutan

Kecepatan Angin m/s	Tegangan v
3	2,66
4	2,8
5	2,75
6	2,86
7	2,82
8	2,93
9	2,99
10	2,95
11	2,88
12	2,83
13	2,9
14	2,98
15	2,95
16	2,93
17	2,91

**Gambar 4.6** Hubungan antara Frekuensi ke Tegangan

Berikut ini hasil perhitungan nilai karakteristik statik alat ukur kecepatan angin berdasarkan data pada Tabel 4.2.

Sensitivitas (dari data pengujian alat) :

- $\text{Sensitivitas} = \frac{6,08-3,8}{5,06-3,2} = 1,225806452$

- *Non-Linieritas*

$$(N(I)) = O(I) - (KI + a)$$

*data yang dihitung adalah data pembacaan kecepatan angin naik

Non – linieritas maksimum per unit

$$= \frac{N}{O_{max} - O_{min}} \times 100\%$$

Dimana :

$$K (\text{sensitivitas}) = 1,37$$

$$a (\text{zero bias}) = 3,8 - (1,225 \times 3,2)$$

$$a (\text{zero bias}) = -0,123$$

$$N (\text{Non Linieritas Maksimum}) = 0,102$$

sehingga :

Non-linieritas maksimum per unit

$$= \frac{0,102}{6,08 - 3,8} \times 100\% = 4,471\%$$

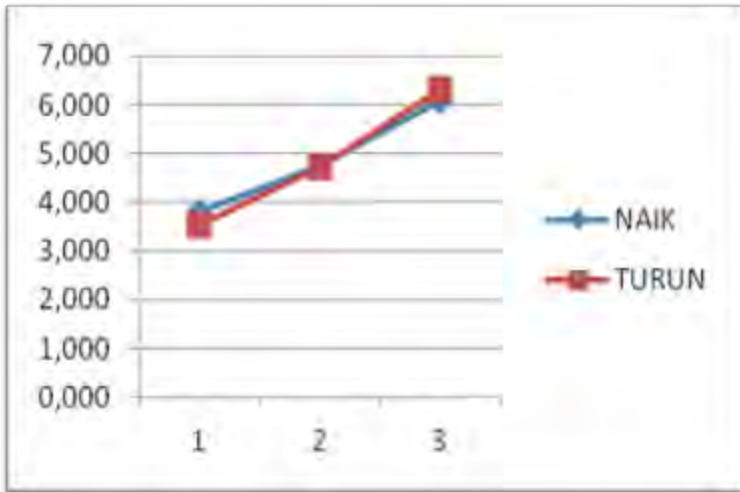
- Histerisis

$$H(I) = O(I)_{I\uparrow} - O(I)_{I\downarrow}, \hat{H} = H(I)_{max} \text{ sehingga :}$$

% maksimum histerisis =

$$= \frac{0,24}{6,08 - 3,8} \times 100\% = 3,947\%$$

Dari histerisis tersebut dapat dijadikan grafik. Berikut ini merupakan grafik histerisis dari pengukuran naik dan turun tersebut.



Gambar 4.7 Histerisis

- Akurasi :

$$A = 1 - |0,201 \times 100\% = 79,934\%$$

Setelah diketahui karakteristik statik dari alat ukur kecepatan angin, langkah berikutnya adalah kalibrasi alat ukur. Yang digunakan sebagai kalibrator adalah *anemometer*. Berikut ini merupakan hasil pengukuran kalibrasi untuk mencari nilai ketidakpastian alat ukur.

Tabel 4.5 Data Kalibrasi

KOREKSI KE-	DI-D ²	X _i *Y _i	X _i ²	SSR
1	1,62129	-1,4720	10,24	0,105795
2	2,73340	-3,2760	15,21	0,519678
3	3,81538	-5,7684	25,6036	1,095747
Jumlah:	8,17007	-10,5164	51,0536	1,721220

Tabel 4.5 Lanjutan

a	-0,2006165
b	0,022321
SIMPANGAN BAKU (STANDAR DEVIASI)	0,952778082
KETIDAKPASTIAN HASIL PENGUKURAN (UA1)	0,301294885
KETIDAKPASTIAN REGRESI (UA2)	0,4638
KETIDAKPASTIAN ALAT STANDAR (UB1)	0,000288675
KETIDAKPASTIAN RESOLUSI (UB2)	0,010604109
KETIDAKPASTIAN GABUNGAN (UC)	0,553212136
DERAJAT KEBEBASAN EFEKTIF (Veff)	15,45834242
FAKTOR CAKUPAN (K)	2,10491
KETIDAKPASTIAN DIPERLUAS (Uexp)	1,164461758

Berikut merupakan perhitungan ketidakpastian alat ukur berdasarkan tabel 4.3..

- Nilai ketidakpastian tipe A :

$$\sigma = \frac{\sqrt{\sum (y_i - \bar{y})^2}}{n - 1}$$

Dimana :

$$\sigma = 0,952778082$$

Sehingga nilai ketidakpastian tipe A adalah :

$$U_{a1} = \frac{\sigma}{\sqrt{n}}$$

$$U_{a1} = \frac{0,952778082}{\sqrt{3}} = 0,550086682$$

- Sedangkan nilai ketidakpastian regresi U_{a2} adalah

$$U_{a2} = \sqrt{\frac{SSR}{n-2}}$$

Dimana :

SSR (*Sum Square Residual*) = $\sum SR$ (*Square Residual*)

SR = R^2 (*Residu*)

Yi (Nilai koreksi) = Pemb. standar (ti) – Pemb. alat (xi)

$$a = \bar{y}_i - (b \times \bar{t}_i)$$

$$b = \frac{n \cdot \sum t_i y_i - \sum y_i \cdot \sum t_i}{n \cdot \sum t_i^2 - (\sum t_i)^2}; \quad t_i = \text{Pemb. standar},$$

yi = Nilai koreksi, n = Jumlah data

$$b = \frac{(10 \times -10,5164) - (4,053333 \times -0,8133)}{(10 \times 51,0536) - (4,053333)^2}$$

$$b = -0,206165$$

Sehingga nilai :

$$a = (-0,8133) - (-0,206165 \times 4,053333)$$

$$a = 0,022321$$

Jadi, persamaan nilai SSR = 1,721220

$$U_{a2} = \sqrt{\frac{1,72122}{10-2}}$$

$$U_{a2} = \mathbf{0,022321}$$

- Nilai ketidakpastian tipe B

Pada ketidakpastian tipe B ini terdapat 2 parameter ketidakpastian, yaitu ketidakpastian Resolusi (U_{B1}) dan ketidakpastian alat standar *anemometer* (U_{B2}). Berikut ini adalah perhitungan ketidakpastian tipe B :

$$U_{B1} = \frac{\frac{1}{2}x_{0,01}}{\sqrt{3}} = 0,002886751$$

$$U_{B2} = \frac{0,022321}{2,10491}$$

- Nilai ketidakpastian kombinasi U_c :

$$U_c = \sqrt{0,301^2 + 0,4638^2 + 0,0028^2 + 0,0106^2}$$

$$U_c = 0,5532$$

Dengan kondisi V atau derajat kebebasan dari kedua tipe ketidakpastian, sebagai berikut :

$V = n-1$, sehingga :

$V1 = 14$; $V2 = 14$; $V3 = \infty$; $V4 = 0$ (berdasarkan table T)

Dengan nilai V_{eff} (Nilai derajat kebebasan efektif) sebagai berikut :

$$V_{eff} = \frac{(0,5532)^4}{(0,30129)^4/9 + (0,4638)^4/9 + (0,0028)^4/ + (0,01)^4/0}$$

$V_{eff} = 15,459$, sehingga bila dihitung dengan interpolasi pada tabel *t-student*, maka menghasilkan nilai k (faktor koreksi) sebesar 2,10491.

Oleh karena itu, hasil nilai ketidakpastian diperluas sebesar :

$$U_{exp} = 2,10491 \times 0,553219593 = 1,164477453$$

Sehingga berdasarkan perhitungan ketidakpastian diperluas diatas, menghasilkan nilai ketidakpastian alat sebesar $\pm 1,164477453$ dengan tingkat kepercayaan 95% dari tabel *T-Student*. Nilai ketidakpastian tersebut akan menjadi acuan untuk pembacaan alat ukur selama alat ukur tersebut digunakan.

4.1.3 Pengujian Alat ukur dan Monitoring Kecepatan Angin pada Turbin Angin tipe Horizontal Axis

Pengujian alat ukur kecepatan angin dengan menggunakan software monitoring visual studio dan penyimpanan data logger MYSQL diperoleh data pembacaan sebagai berikut:



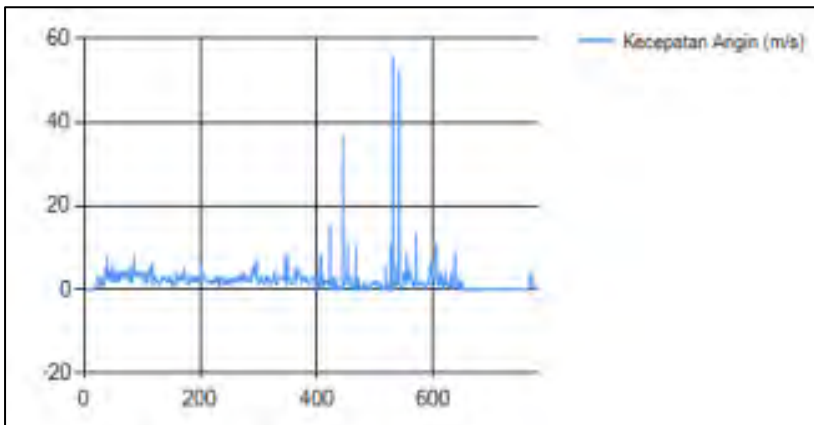
Gambar 4.8 *Interface* Monitoring Kecepatan Angin

Tabel 4.6 Database Monitoring Kecepatan Angin pada tanggal 19 Juli 2016

No	Waktu		Kecepatan Angin
1	2016/07/19	22:09:41	1.199
2	2016/07/19	22:09:46	2.797
3	2016/07/19	22:09:51	2.797
4	2016/07/19	22:09:56	2.598
5	2016/07/19	22:10:01	7.993
6	2016/07/19	22:10:06	3.797
7	2016/07/19	22:10:11	1.998

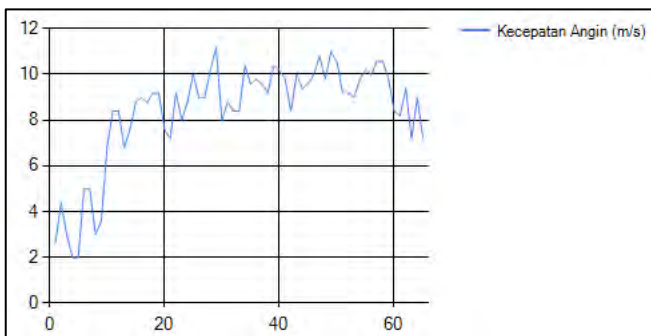
Tabel 4.6 Lanjutan

No	Waktu		Kecepatan Angin
8	2016/07/19	22:10:16	3.197
9	2016/07/19	22:10:21	4.596
10	2016/07/19	22:10:26	3.797
11	2016/07/19	22:10:31	3.996
12	2016/07/19	22:10:36	3.797
13	2016/07/19	22:10:41	2.997
14	2016/07/19	22:10:46	8.392
15	2016/07/19	22:10:51	3.197
16	2016/07/19	22:10:56	4.196
17	2016/07/19	22:11:01	2.997

**Gambar 4.9** Grafik Perbandingan kenaikan kecepatan angin terhadap waktu pada tanggal 19 Juli 2016

Tabel 4.7 Database Monitoring Kecepatan Angin pada tanggal 25 Juli 2016

No	Waktu	Kecepatan Angin
1	25/07/2016 18:36	2.598
2	25/07/2016 18:36	4.396
3	25/07/2016 18:36	2.997
4	25/07/2016 18:36	1.998
5	25/07/2016 18:36	1.998
6	25/07/2016 18:36	4.995
7	25/07/2016 18:36	4.995
8	25/07/2016 18:36	2.997
9	25/07/2016 18:36	3.597
10	25/07/2016 18:36	6.794
11	25/07/2016 18:36	8.392
12	25/07/2016 18:36	8.392
13	25/07/2016 18:36	6.794
14	25/07/2016 18:37	7.593
15	25/07/2016 18:37	8.792



Gambar 4.10 Grafik Perbandingan kenaikan kecepatan angin terhadap waktu pada tanggal 25 Juli 2016

4.2 Analisa Data

Pada Tugas Akhir ini yaitu tentang Rancang Bangun Sistem Monitoring Kecepatan Angin Berbasis Mikrokontroler ATmega32 pada Turbin Angin Tipe Horizontal Axis. Pengukuran kecepatan angin dilakukan dengan menggunakan modul sensor optocoupler. Hasil pengukuran ditampilkan pada LCD 2 x 16 dan dilakukan *interfacing* dengan menggunakan *software visual basic* yang telah terhubung dengan *database MySQL* dan *data logger* dengan MMC.

Prinsip kerja dari alat ukur kecepatan angin adalah *sensing element* yang dilakukan oleh modul sensor optocoupler. *Output* dari modul sensor optocoupler adalah pencacahan *disk rotary*, pencacahan tersebut menghasilkan pulsa 0 dan 1 yakni sinyal PWM. Kemudian sinyal PWM diolah dengan mikrokontroler ATmega32. *Output* dari ATmega32 sudah berupa data visual yang dapat dilihat melalui LCD. Dari tabel 4.3 dapat diketahui bahwa ketidakpastian diperluas (U_{exp}) = 1,164461758. Hasil akhir dari pengukuran kecepatan angin ini, semisal didapat nilai kecepatan angin 6,594 m/s, maka ditulis $(6,594 \pm 1,164461758)$ m/s. Ketidakpastian diperluas tersebut didapatkan dengan menggunakan tingkat kepercayaan 95%. Akurasi dari alat ukur kecepatan angin ini adalah 79,934% dengan kesalahan 20,066%. *Range* pengukuran yang dapat dilakukan 8-12 Volt/ 3,2 – 5,06 m/s pada standar anemometer dan bantuan *heatsink* DC.

Setelah dilakukan pengujian terhadap alat ukur kecepatan angin, maka alat ukur tersebut digunakan untuk mengukur kecepatan angin di ruangan terbuka. Pengujian tersebut dilakukan dengan mengukur nilai tegangan keluaran dari mikrokontroler pada minimum system ATmega32 dan sensor, yang didapat data berupa penurunan tegangan, hal ini terjadi karena pengaruh dari kabel dan rugi-rugi tegangan sumber. Kegiatan pengukuran tersebut dapat dimonitoring, dengan menggunakan Visual Studio 2013 sebagai *display*, MySQL sebagai penyimpan data monitoring, dan *data logger* dengan MMC. Besarnya kecepatan angin dapat dilihat secara *real time*. Monitoring kecepatan angin

dilakukan di pesisir Pantai Kenjeran Surabaya. Berdasarkan data yang tersimpan di dalam *database* hasil monitoring menunjukkan bahwa kecepatan angin yang terjadi tanggal 19 Juli 2016 pukul 22:09:23 – 22:19:53 memiliki rata-rata kecepatan 2,554 m/s, dan 25 Juli 2016 pukul 18:36:47 – 18:39:53 memiliki rata-rata kecepatan 8,409 m/s.

LAMPIRAN A
(DATABASE MONITORING)

Tabel A2. *Database Monitoring Kecepatan Angin pada tanggal 19 Juli 2016*

No	Waktu		Kecepatan Angin
1	2016/07/19	22:09:41	1.199
2	2016/07/19	22:09:46	2.797
3	2016/07/19	22:09:51	2.797
4	2016/07/19	22:09:56	2.598
5	2016/07/19	22:10:01	7.993
6	2016/07/19	22:10:06	3.797
7	2016/07/19	22:10:11	1.998
8	2016/07/19	22:10:16	3.197
9	2016/07/19	22:10:21	4.596
10	2016/07/19	22:10:26	3.797
11	2016/07/19	22:10:31	3.996
12	2016/07/19	22:10:36	3.797
13	2016/07/19	22:10:41	2.997
14	2016/07/19	22:10:46	8.392
15	2016/07/19	22:10:51	3.197
16	2016/07/19	22:10:56	4.196
17	2016/07/19	22:11:01	2.997
18	2016/07/19	22:11:06	1.399
19	2016/07/19	22:11:11	1.998
20	2016/07/19	22:11:16	3.197
21	2016/07/19	22:11:21	2.997
22	2016/07/19	22:11:26	1.798
23	2016/07/19	22:11:31	1.599
24	2016/07/19	22:11:36	2.598

No	Waktu		Kecepatan Angin
25	2016/07/19	22:11:42	3.197
26	2016/07/19	22:11:47	2.398
27	2016/07/19	22:11:52	1.998
28	2016/07/19	22:11:57	1.798
29	2016/07/19	22:12:02	1.798
30	2016/07/19	22:12:07	2.797
31	2016/07/19	22:12:12	2.997
32	2016/07/19	22:12:17	3.397
33	2016/07/19	22:12:22	2.198
34	2016/07/19	22:12:27	1.998
35	2016/07/19	22:12:32	2.797
36	2016/07/19	22:12:37	3.197
37	2016/07/19	22:12:42	4.196
38	2016/07/19	22:12:47	3.197
39	2016/07/19	22:12:52	2.198
40	2016/07/19	22:13:02	2.398
41	2016/07/19	22:13:07	2.198
42	2016/07/19	22:13:12	1.599
43	2016/07/19	22:13:17	2.598
44	2016/07/19	22:13:22	2.797
45	2016/07/19	22:13:27	2.398
46	2016/07/19	22:13:32	1.998
47	2016/07/19	22:13:37	2.198
48	2016/07/19	22:13:42	2.997
49	2016/07/19	22:13:47	1.599
50	2016/07/19	22:13:52	3.197
51	2016/07/19	22:14:02	2.997
52	2016/07/19	22:14:07	2.797
53	2016/07/19	22:14:12	3.597

No	Waktu	Kecepatan Angin
54	2016/07/19 22:14:17	2.997

Tabel A2. *Database* Monitoring Kecepatan Angin pada tanggal 25 Juli 2016

No	Waktu	Kecepatan Angin
1	25/07/2016 18:36	2.598
2	25/07/2016 18:36	4.396
3	25/07/2016 18:36	2.997
4	25/07/2016 18:36	1.998
5	25/07/2016 18:36	1.998
6	25/07/2016 18:36	4.995
7	25/07/2016 18:36	4.995
8	25/07/2016 18:36	2.997
9	25/07/2016 18:36	3.597
10	25/07/2016 18:36	6.794
11	25/07/2016 18:36	8.392
12	25/07/2016 18:36	8.392
13	25/07/2016 18:36	6.794
14	25/07/2016 18:37	7.593
15	25/07/2016 18:37	8.792
16	25/07/2016 18:37	8.992
17	25/07/2016 18:37	8.792
18	25/07/2016 18:37	9.192
19	25/07/2016 18:37	9.192
20	25/07/2016 18:37	7.593
21	25/07/2016 18:37	7.193
22	25/07/2016 18:37	9.192
23	25/07/2016 18:37	7.993

No	Waktu	Kecepatan Angin
24	25/07/2016 18:37	8.792
25	25/07/2016 18:37	9.991
26	25/07/2016 18:37	8.992
27	25/07/2016 18:37	8.992
28	25/07/2016 18:37	10.191
29	25/07/2016 18:37	11.19
30	25/07/2016 18:37	7.993
31	25/07/2016 18:37	8.792
32	25/07/2016 18:37	8.392
33	25/07/2016 18:37	8.392
34	25/07/2016 18:37	10.391
35	25/07/2016 18:37	9.591
36	25/07/2016 18:37	9.791
37	25/07/2016 18:37	9.591
38	25/07/2016 18:37	9.192
39	25/07/2016 18:37	10.391
40	25/07/2016 18:37	10.191
41	25/07/2016 18:37	9.791
42	25/07/2016 18:37	8.392
43	25/07/2016 18:37	9.991
44	25/07/2016 18:37	9.391
45	25/07/2016 18:37	9.591
46	25/07/2016 18:37	9.991
47	25/07/2016 18:37	10.79
48	25/07/2016 18:37	9.791
49	25/07/2016 18:37	10.99
50	25/07/2016 18:37	10.59
51	25/07/2016 18:37	9.192

52	25/07/2016 18:37	9.192
53	25/07/2016 18:37	8.992
54	25/07/2016 18:37	9.791
55	25/07/2016 18:37	10.191
56	25/07/2016 18:37	9.991
57	25/07/2016 18:37	10.59
58	25/07/2016 18:37	10.59
59	25/07/2016 18:37	9.791
60	25/07/2016 18:37	8.392
61	25/07/2016 18:37	8.193
62	25/07/2016 18:37	9.391
63	25/07/2016 18:37	7.193
64	25/07/2016 18:37	8.992
65	25/07/2016 18:37	7.193
66	25/07/2016 18:37	8.193
67	25/07/2016 18:37	7.593
68	25/07/2016 18:37	6.794
69	25/07/2016 18:37	8.992
70	25/07/2016 18:37	6.594
71	25/07/2016 18:37	7.193
72	25/07/2016 18:37	9.391
73	25/07/2016 18:38	9.591
74	25/07/2016 18:38	9.591
75	25/07/2016 18:38	7.393
76	25/07/2016 18:38	8.992
77	25/07/2016 18:38	9.391
78	25/07/2016 18:38	7.793
79	25/07/2016 18:38	9.192
80	25/07/2016 18:38	10.191
81	25/07/2016 18:38	9.192

No	Waktu	Kecepatan Angin
82	25/07/2016 18:38	7.793
83	25/07/2016 18:38	7.393
84	25/07/2016 18:38	8.392
85	25/07/2016 18:38	8.992
86	25/07/2016 18:38	9.391
87	25/07/2016 18:38	7.993
88	25/07/2016 18:38	8.193
89	25/07/2016 18:38	7.993
90	25/07/2016 18:38	7.593
91	25/07/2016 18:38	8.392
92	25/07/2016 18:38	8.792
93	25/07/2016 18:38	9.991
94	25/07/2016 18:38	7.993
95	25/07/2016 18:38	7.393
96	25/07/2016 18:38	8.592
97	25/07/2016 18:38	9.591
98	25/07/2016 18:38	8.193
99	25/07/2016 18:38	8.992
100	25/07/2016 18:38	6.794
101	25/07/2016 18:38	7.193
102	25/07/2016 18:38	8.392
103	25/07/2016 18:38	8.992
104	25/07/2016 18:38	8.592
105	25/07/2016 18:38	9.192
106	25/07/2016 18:38	6.994
107	25/07/2016 18:38	10.391
108	25/07/2016 18:38	10.191
109	25/07/2016 18:38	9.391

No	Waktu	Kecepatan Angin
110	25/07/2016 18:38	8.193
111	25/07/2016 18:38	8.392
112	25/07/2016 18:38	7.993
113	25/07/2016 18:38	8.992
114	25/07/2016 18:38	9.192
115	25/07/2016 18:38	10.79
116	25/07/2016 18:38	8.992
117	25/07/2016 18:38	7.593
118	25/07/2016 18:38	7.993
119	25/07/2016 18:38	8.193
120	25/07/2016 18:38	7.593
121	25/07/2016 18:38	8.193
122	25/07/2016 18:38	9.791
123	25/07/2016 18:38	8.792
124	25/07/2016 18:38	9.192
125	25/07/2016 18:38	10.191
126	25/07/2016 18:38	7.793
127	25/07/2016 18:38	9.591
128	25/07/2016 18:38	9.391
129	25/07/2016 18:38	7.593
130	25/07/2016 18:38	8.392
131	25/07/2016 18:38	8.592
132	25/07/2016 18:38	8.992
133	25/07/2016 18:39	8.992
134	25/07/2016 18:39	9.192
135	25/07/2016 18:39	8.392
136	25/07/2016 18:39	9.192
137	25/07/2016 18:39	10.59

No	Waktu	Kecepatan Angin
138	25/07/2016 18:39	7.593
139	25/07/2016 18:39	8.392
140	25/07/2016 18:39	8.193
141	25/07/2016 18:39	9.192
142	25/07/2016 18:39	10.391
143	25/07/2016 18:39	9.591
144	25/07/2016 18:39	8.592
145	25/07/2016 18:39	9.192
146	25/07/2016 18:39	9.991
147	25/07/2016 18:39	8.792
148	25/07/2016 18:39	8.792
149	25/07/2016 18:39	8.992
150	25/07/2016 18:39	10.191
151	25/07/2016 18:39	9.791
152	25/07/2016 18:39	10.191
153	25/07/2016 18:39	9.991
154	25/07/2016 18:39	9.791
155	25/07/2016 18:39	7.393
156	25/07/2016 18:39	8.592
157	25/07/2016 18:39	7.993
158	25/07/2016 18:39	9.791
159	25/07/2016 18:39	9.591
160	25/07/2016 18:39	8.792
161	25/07/2016 18:39	10.391
162	25/07/2016 18:39	8.992
163	25/07/2016 18:39	8.392
164	25/07/2016 18:39	8.592
165	25/07/2016 18:39	8.193

No	Waktu	Kecepatan Angin
166	25/07/2016 18:39	8.992
167	25/07/2016 18:39	8.992
168	25/07/2016 18:39	9.991
169	25/07/2016 18:39	10.391
170	25/07/2016 18:39	10.391
171	25/07/2016 18:39	7.793
172	25/07/2016 18:39	8.392
173	25/07/2016 18:39	7.793
174	25/07/2016 18:39	8.193
175	25/07/2016 18:39	8.592
176	25/07/2016 18:39	7.193
177	25/07/2016 18:39	9.192
178	25/07/2016 18:39	8.792
179	25/07/2016 18:39	9.192
180	25/07/2016 18:39	8.992
181	25/07/2016 18:39	8.592
182	25/07/2016 18:39	7.793
183	25/07/2016 18:39	9.591
184	25/07/2016 18:39	9.391
185	25/07/2016 18:39	9.991
186	25/07/2016 18:39	7.793
187	25/07/2016 18:39	9.192
188	25/07/2016 18:39	8.992
189	25/07/2016 18:39	8.992
190	25/07/2016 18:39	6.194
191	25/07/2016 18:39	9.591
192	25/07/2016 18:40	9.391
193	25/07/2016 18:40	8.992

No	Waktu	Kecepatan Angin
194	25/07/2016 18:40	10.391
195	25/07/2016 18:40	8.592
196	25/07/2016 18:40	8.592
197	25/07/2016 18:40	7.793
198	25/07/2016 18:40	9.391
199	25/07/2016 18:40	9.192
200	25/07/2016 18:40	8.592
201	25/07/2016 18:40	9.391
202	25/07/2016 18:40	9.391
203	25/07/2016 18:40	9.391
204	25/07/2016 18:40	8.992
205	25/07/2016 18:40	10.191
206	25/07/2016 18:40	9.591
207	25/07/2016 18:40	9.391
208	25/07/2016 18:40	11.19
209	25/07/2016 18:40	10.59
210	25/07/2016 18:40	10.191
211	25/07/2016 18:40	8.992
212	25/07/2016 18:40	8.392
213	25/07/2016 18:40	8.193
214	25/07/2016 18:40	9.391
215	25/07/2016 18:40	8.792
216	25/07/2016 18:40	8.392
217	25/07/2016 18:40	8.992
218	25/07/2016 18:40	9.192
219	25/07/2016 18:40	8.592
220	25/07/2016 18:40	8.592
221	25/07/2016 18:40	8.792

No	Waktu	Kecepatan Angin
222	25/07/2016 18:40	9.391
223	25/07/2016 18:40	9.192
224	25/07/2016 18:40	9.791
225	25/07/2016 18:40	9.791
226	25/07/2016 18:40	7.393
227	25/07/2016 18:40	8.392
228	25/07/2016 18:40	8.792
229	25/07/2016 18:40	7.393
230	25/07/2016 18:40	8.392
231	25/07/2016 18:40	8.193
232	25/07/2016 18:40	8.592
233	25/07/2016 18:40	6.794
234	25/07/2016 18:40	5.595
235	25/07/2016 18:40	3.197
236	25/07/2016 18:40	3.397
237	25/07/2016 18:40	7.393
238	25/07/2016 18:40	8.193
239	25/07/2016 18:40	7.993
240	25/07/2016 18:40	8.392
241	25/07/2016 18:40	7.793
242	25/07/2016 18:40	7.793
243	25/07/2016 18:40	7.793
244	25/07/2016 18:40	8.792
245	25/07/2016 18:40	7.793
246	25/07/2016 18:40	7.993
247	25/07/2016 18:40	6.994
248	25/07/2016 18:40	6.394
249	25/07/2016 18:40	5.395

No	Waktu	Kecepatan Angin
250	25/07/2016 18:40	5.595
251	25/07/2016 18:40	8.193
252	25/07/2016 18:41	8.392
253	25/07/2016 18:41	8.193
254	25/07/2016 18:41	8.792
255	25/07/2016 18:41	9.791
256	25/07/2016 18:41	10.59
257	25/07/2016 18:41	10.99
258	25/07/2016 18:41	9.591
259	25/07/2016 18:41	8.392
260	25/07/2016 18:41	7.593
261	25/07/2016 18:41	8.592
262	25/07/2016 18:41	7.993
263	25/07/2016 18:41	7.393
264	25/07/2016 18:41	6.394
265	25/07/2016 18:41	7.793
266	25/07/2016 18:41	7.193
267	25/07/2016 18:41	4.796
268	25/07/2016 18:41	8.592
269	25/07/2016 18:41	8.392
270	25/07/2016 18:41	7.193
271	25/07/2016 18:41	8.592
272	25/07/2016 18:41	8.592
273	25/07/2016 18:41	7.393
274	25/07/2016 18:41	7.593
275	25/07/2016 18:41	6.794
276	25/07/2016 18:41	7.393
277	25/07/2016 18:41	5.595

No	Waktu	Kecepatan Angin
278	25/07/2016 18:41	8.792
279	25/07/2016 18:41	7.793
280	25/07/2016 18:41	8.193

LAMPIRAN B

(Listing Program di Mikrokontroler dan Microsoft Visual Studio 2013)

B.1 Listing Program ATmega32

/******

*

This program was produced by the
CodeWizardAVR V2.05.3 Standard
Automatic Program Generator
© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>

Project :

Version :

Date : 02/08/2016

Author : Shinta

Company : its

Comments:

Chip type : ATmega32A

Program type : Application

AVR Core Clock frequency: 4,000000 MHz

Memory model : Small

External RAM size : 0

Data Stack size : 512

/

```
#include <mega32a.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <ff.h>
```

```
#include <sdcard.h>
```

```
#include <delay.h>
```

```

#define _ALTERNATE_PUTCHAR
// Alphanumeric LCD functions
#include <alcd.h>

// Declare your global variables here
#define LCD_RS_HI PORTA|=(1<<7)
#define LCD_RS_LO PORTA&=~(1<<7)

#define LCD_EN_HI PORTA|=(1<<6)
#define LCD_EN_LO PORTA&=~(1<<6)

#define LCD_D4_HI PORTA|=(1<<5)
#define LCD_D4_LO PORTA&=~(1<<5)
#define LCD_D5_HI PORTA|=(1<<4)
#define LCD_D5_LO PORTA&=~(1<<4)
#define LCD_D6_HI PORTA|=(1<<3)
#define LCD_D6_LO PORTA&=~(1<<3)
#define LCD_D7_HI PORTA|=(1<<2)
#define LCD_D7_LO PORTA&=~(1<<2)

//=====

/* Timer1 overflow interrupt frequency [Hz] */
#define T2_OVF_FREQ 100
/* Timer1 clock prescaler value */
#define T2_PRESC 1024L
/* Timer1 initialization value after overflow */
#define T2_INIT (0x10000L-
(_MCU_CLOCK_FREQUENCY_/(T2_PRESC*T2_OVF_FREQ
)))

/* FAT function result */
FRESULT res;
/* number of bytes written/read to the file */
unsigned int nbytes;

```

```

/* will hold the information for logical drive 0: */
FATFS fat;
/* will hold the file information */
FIL file;
/* will hold file attributes, time stamp information */
FILINFO finfo;
char buffer[100];
char filename[30];
float datakecepatan;
int frekuensi=0;
float rpm;
char temp[10];
unsigned int panjangdatammc;
unsigned char status_tulis,status_create;
char c[10];

//=====

=====

#define line1 0x00
#define line2 0x40
//=====
void LCD_STROBE()
{
    LCD_EN_HI;
    LCD_EN_LO;
}

void lcd_write(unsigned char datalcd)
{
    if(datalcd & 0x80) {LCD_D7_HI;} else {LCD_D7_LO;}
    if(datalcd & 0x40) {LCD_D6_HI;} else {LCD_D6_LO;}
    if(datalcd & 0x20) {LCD_D5_HI;} else {LCD_D5_LO;}
    if(datalcd & 0x10) {LCD_D4_HI;} else {LCD_D4_LO;}
    LCD_STROBE();
}

```

```
if(datalcd & 0x08) {LCD_D7_HI;} else {LCD_D7_LO;}
if(datalcd & 0x04) {LCD_D6_HI;} else {LCD_D6_LO;}
if(datalcd & 0x02) {LCD_D5_HI;} else {LCD_D5_LO;}
if(datalcd & 0x01) {LCD_D4_HI;} else {LCD_D4_LO;}
LCD_STROBE();
delay_ms(5);
}
```

```
void lcd_clrscr()
{
    LCD_RS_LO;
    lcd_write(0x1);
    delay_ms(2);
}
```

```
void lcd_putchar(char c)
{
    LCD_RS_HI;
    lcd_write(c);
}
```

```
void lcd_goto(unsigned char poslcd)
{
    LCD_RS_LO;
    lcd_write(0x80 + poslcd);
}
```

```
void init_lcd()
{
    LCD_RS_LO;
    delay_ms(15);
    LCD_D4_HI;
    LCD_D5_HI;
    LCD_STROBE();
    delay_ms(5);
}
```

```

LCD_STROBE();
delay_us(100);
LCD_STROBE();
delay_ms(5);
LCD_D4_LO;
LCD_STROBE();
delay_us(40);
lcd_write(0x28);
lcd_write(0x0C);
lcd_write(0x06);
lcd_write(0x01);
delay_ms(10);
}

```

```

void lcd_putsf(char flash *progmem_s)
{
    char c,iscan;
    iscan=0;
    for(;;)
    {
        c = *progmem_s++;
        if(c==0)break;
        lcd_putchar(char c);
        iscan++;
    }
}

```

```

void putchar(unsigned char c)
{
    LCD_RS_HI;
    lcd_write(c);
}
//=====
=====
//mmc
//

```



```

void timer_mmc()
{
    //TIMER
    TCCR1A=0x00;
    TCCR1B=(1<<CS12)|(1<<CS10);
    TCNT1H=T2_INIT>>8;
    TCNT1L=T2_INIT&0xFF;
    TIMSK=1<<TOIE1;
}

void stop_timer_mmc()
{
    //TIMER
    TCCR1A=0x00;
    TCCR1B=0;
    TCNT1H=T2_INIT>>8;
    TCNT1L=T2_INIT&0xFF;
    TIMSK&=~(1<<TOIE1);

}

void error(FRESULT res)
{
    if ((res>=FR_DISK_ERR) && (res<=FR_TIMEOUT))
    {
        //poutput=USART1;
        //printf("ERROR: %p\r\n",error_msg[res]);
    }
}

void display_status(char *file_name)
{
    if ((res=f_stat(file_name,&finfo))==FR_OK)
    {

```

```

}
}
//=====
==
void inisialisasi_fat(void)
{
    timer_mmc();
    for(;;)
    {
        if ((res=f_mount(0,&fat))==FR_OK)
        {
            break;
        }
        else
        {
            error(res);
        }
    }
    stop_timer_mmc();
}

void reset_filename(void)
{
    unsigned char i_reset;
    for(i_reset=0;i_reset<30;i_reset++)
    {
        filename[i_reset]=0;
    }
}
//=====
=====
void file_data_kecepatan(void)
{
    filename[0]='0';
    filename[1]='.';
    filename[2]='k';

```

```

filename[3]='e';
filename[4]='c';
filename[5]='e';
filename[6]='p';
filename[7]='a';
filename[8]='t';
filename[9]='a';
filename[10]='n';
filename[11]='.';
filename[12]='t';
filename[13]='x';
filename[14]='t';
}

```

```

void file_baru(void)
{
    unsigned char ulang;
    status_create=0;
    for(ulang=0;ulang<10;ulang++)
    {
        reset_filename();
        file_data_kecepatan();
        timer_mmc();

        if
        ((res=f_open(&file,filename,FA_CREATE_NEW))==FR_OK)sta
tus_create=1;
        else status_create=0;

        if(status_create==1)
        {
            if ((res=f_close(&file))==FR_OK)status_create=1;
            else status_create=0;
        }

        if(status_create==1)break;
    }
}

```

```

    }
    stop_timer_mmc();
}

//=====
=====

void isi_file(void)
{
    unsigned ulang;
    status_tulis=0;

    reset_filename();
    file_data_kecepatan();
    display_status(filename);

    for(ulang=0;ulang<10;ulang++)
    {
        timer_mmc();
        if
        ((res=f_open(&file,filename,FA_WRITE))==FR_OK)status_tulis
        =1;
        else status_tulis=0;

        if(status_tulis==1)
        {
            if
            ((res=f_lseek(&file,finfo.fsize))==FR_OK)status_tulis=1;
            else status_tulis=0;
        }

        if(status_tulis==1)
        {
            if
            ((res=f_write(&file,buffer,panjangdatammc+1,&nbytes))==FR_O
            K)status_tulis=1;

```

```

        else status_tulis=0;
    }

    if(status_tulis==1)
    {
        if ((res=f_close(&file))==FR_OK)status_tulis=1;
        else status_tulis=0;
    }

    if(status_tulis==1)break;
    delay_ms(500);
}
stop_timer_mmc();
}

void copy_kecepatan_mmc()
{
    panjangdatammc++;
    buffer[panjangdatammc]='k';
    panjangdatammc++;
    buffer[panjangdatammc]='e';
    panjangdatammc++;
    buffer[panjangdatammc]='c';
    panjangdatammc++;
    buffer[panjangdatammc]='e';
    panjangdatammc++;
    buffer[panjangdatammc]='p';
    panjangdatammc++;
    buffer[panjangdatammc]='a';
    panjangdatammc++;
    buffer[panjangdatammc]='t';
    panjangdatammc++;
    buffer[panjangdatammc]='a';
    panjangdatammc++;
    buffer[panjangdatammc]='n';
    panjangdatammc++;
}

```

```

    buffer[panjangdatammc]='.';
    panjangdatammc++;
}

```

```

// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{
// Place your code here
if(PIND.2==0)
{frekuensi ++;}
else if(PIND.2==1)
{frekuensi --;}
}

```

// Standard Input/Output functions

```

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
// Reinitialize Timer1 value
TCNT1H=0xF0BD >> 8;
TCNT1L=0xF0BD & 0xff;
// Place your code here
rpm=(float)frekuensi*60/22;
frekuensi=0;
datakecepatan=(float)rpm*((2*3.14*0.07)/60)*10;
}

```

```

// Timer2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
// Place your code here
TCNT1H=T2_INIT>>8;
TCNT1L=T2_INIT&0xFF;
disk_timerproc();
}

```

```
}
```

```
// SPI functions
#include <spi.h>
```

```
void main(void)
{
// Declare your local variables here
```

```
// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T
State1=T State0=T
PORTA=0x00;
DDRA=0x00;
```

```
// Port B initialization
// Func7=Out Func6=In Func5=Out Func4=Out Func3=In
Func2=In Func1=In Func0=In
// State7=0 State6=T State5=0 State4=0 State3=T State2=T
State1=T State0=T
PORTB=0x00;
DDRB=0xB0;
```

```
// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T
State1=T State0=T
PORTC=0x00;
DDRC=0x00;
```

```
// Port D initialization
```

```
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In  
Func1=In Func0=In  
// State7=T State6=T State5=T State4=T State3=T State2=T  
State1=T State0=T  
PORTD=0x00;  
DDRD=0x00;
```

```
// Timer/Counter 0 initialization  
// Clock source: System Clock  
// Clock value: Timer 0 Stopped  
// Mode: Normal top=0xFF  
// OC0 output: Disconnected  
TCCR0=0x00;  
TCNT0=0x00;  
OCR0=0x00;
```

```
// Timer/Counter 1 initialization  
// Clock source: System Clock  
// Clock value: 3,906 kHz  
// Mode: Ph. correct PWM top=0x03FF  
// OC1A output: Discon.  
// OC1B output: Discon.  
// Noise Canceler: Off  
// Input Capture on Falling Edge  
// Timer1 Overflow Interrupt: On  
// Input Capture Interrupt: Off  
// Compare A Match Interrupt: Off  
// Compare B Match Interrupt: Off  
TCCR1A=0x03;  
TCCR1B=0x05;  
TCNT1H=0xF0;  
TCNT1L=0xBD;  
ICR1H=0x00;  
ICR1L=0x00;  
OCR1AH=0x00;  
OCR1AL=0x00;
```



```

OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 4000,000 kHz
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x01;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: On
// INT0 Mode: Falling Edge
// INT1: Off
// INT2: Off
GICR|=0x40;
MCUCR=0x02;
MCUCSR=0x00;
GIFR=0x40;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x44;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x06;
UBRRH=0x00;

```

```

UBRRL=0x19;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIO=0x00;

// ADC initialization
// ADC disabled
ADCSRA=0x00;

// SPI initialization
// SPI Type: Master
// SPI Clock Rate: 1000,000 kHz
// SPI Clock Phase: Cycle Start
// SPI Clock Polarity: Low
// SPI Data Order: MSB First
SPCR=0x50;
SPSR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD
menu:
// RS - PORTA Bit 0
// RD - PORTA Bit 1
// EN - PORTA Bit 2
// D4 - PORTA Bit 4
// D5 - PORTA Bit 5
// D6 - PORTA Bit 6
// D7 - PORTA Bit 7

```

```

// Characters/line: 16
lcd_init(16);
lcd_goto(line1);
printf("create file");

// Global enable interrupts
#asm("sei")
inisialisasi_fat();
file_baru();
lcd_clrscr();
lcd_gotoxy(0,0);
lcd_putsf("Kecepatan Angin :)");

while (1)
{
    // Place your code here
    ftoa(datakecepatan,3,temp);
    lcd_gotoxy(0,1);
    lcd_puts(temp);
    lcd_gotoxy(0,9);
    lcd_putsf(" m/s");
    puts(temp);
    delay_ms(1000);
    {
        lcd_clrscr();
        lcd_goto(line1);
        printf("save data");
        copy_kecepatan_mmc();
        isi_file();
        delay_ms(1000);
        lcd_clrscr();
    }
}
}

```

B.2 Listing program Visual Basic 2013

```
Imports System.IO.Ports
Imports MySql.Data.MySqlClient
Public Class SINTA
    Private sqlconn As New MySqlConnection
    Private sqlcomm As MySqlCommand
    Private WithEvents myserial As New SerialPort
    Private kec As Double
    Private Sub SINTA_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
        sqlconn.ConnectionString = "Server=127.0.0.1;Database=ta
sinta;Uid=root;Pwd=;"
        Try
            sqlconn.Open()
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
        getports()
    End Sub
    Sub getports()
        ComboBox1.Items.Clear()
        For Each i As String In My.Computer.Ports.SerialPortNames
            ComboBox1.Items.Add(i)
        Next
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs)
Handles Button1.Click
        If myserial.IsOpen Then
            myserial.Close()
        If Not myserial.IsOpen Then
            Button1.Text = "Connect"
        End If
    Else
        myserial.BaudRate = 9600
    Try
```

```

        myserial.PortName = ComboBox1.Text
        myserial.Open()
    Catch ex As Exception
        MsgBox(ex.Message)
    End Try
    If myserial.IsOpen Then
        Button1.Text = "Disconnect"
    End If
End If
End Sub

Private Sub myserial_DataReceived(sender As Object, e As
SerialDataReceivedEventArgs) Handles myserial.DataReceived
    Dim baca As String = myserial.ReadLine
    Me.Invoke(New myDel(AddressOf olahdata), baca)
End Sub

Delegate Sub myDel(ByVal [data] As String)
Sub olahdata(ByVal [DATA] As String)
    kec = Val(DATA)
    Chart1.Series("Kecepatan Angin (m/s)").Points.AddY(kec)
    TextBox1.Text = kec.ToString
    simpan("INSERT INTO `kecepatan angin`(`nomor`,
`waktu`, `kecepatan angin`) VALUES (" + " +
Now.ToString("yyyy/MM/dd HH:mm:ss") + " + " + kec.ToString
+ " + ")")
End Sub

Private Sub ComboBox1_Click(sender As Object, e As
EventArgs) Handles ComboBox1.Click
    getports()
End Sub

Sub simpan(ByVal query As String)
    sqlcomm = New MySqlCommand
    Try
        With sqlcomm
            .Connection = sqlconn
            .CommandText = query
            .ExecuteNonQuery()
        End With
    End Try
End Sub

```

```
        End With
    Catch ex As Exception
        MsgBox(ex.Message)
    End Try
End Sub
End Class
```

LAMPIRAN D

Tabel D.1. Degree of Freedom

Degree of freedom V	Probabilitas / Tingkat kepercayaan (%)			
	68,27 %	90 %	95%	99%
1	1,84	6,31	12,71	63,66
2	1,32	2,92	4,30	9,92
3	1,20	2,35	3,18	5,84
4	1,14	2,13	2,78	4,60
5	1,11	2,02	2,57	4,03
6	1,09	1,94	2,45	3,71
7	1,08	1,89	2,36	3,50
8	1,07	1,86	2,31	3,36
9	1,06	1,83	2,26	3,25
10	1,05	1,81	2,23	3,17
11	1,05	1,80	2,20	3,11
12	1,04	1,78	2,18	3,05
13	1,04	1,77	2,16	3,01
14	1,04	1,76	2,14	2,98
15	1,03	1,75	2,13	2,95
16	1,03	1,75	2,12	2,92
17	1,03	1,74	2,11	2,90
18	1,03	1,73	2,10	2,88
19	1,03	1,73	2,09	2,86
20	1,03	1,72	2,09	2,85
25	1,02	1,71	2,06	2,79
30	1,02	1,70	2,04	2,75
35	1,02	1,70	2,03	2,72
40	1,02	1,68	2,02	2,70

45	1,02	1,68	2,01	2,69
50	1,01	1,68	2,01	2,68
100	1,005	1,660	1,984	2,626
∞	1	1,645	1.960	2,576

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pada Tugas Akhir Rancang Bangun Sistem Monitoring Kecepatan Angin Berbasis Mikrokontroler Atmega32 pada Turbin Angin Tipe Horizontal Axis dapat disimpulkan, yaitu :

- a. Sistem monitoring kecepatan angin yang dirancang dapat diaplikasikan pada turbin angin tipe horizontal axis.
- b. Sistem monitoring kecepatan angin memiliki nilai ketidakpastian diperluas (U_{exp}) sebesar $\pm 1,1644$ dengan tingkat kepercayaan 95%. Akurasi alat ukur kecepatan angin adalah 79,934%
- c. Hasil dari monitoring kecepatan angin menyatakan bahwa area turbin angin tipe horizontal axis pada tanggal 19 Juli 2016 pukul 22:09:23 – 22:19:53 memiliki rata-rata kecepatan 2,554 m/s, dan 25 Juli 2016 pukul 18:36:47 – 18:39:53 memiliki rata-rata kecepatan 8,409 m/s.

5.2 Saran

Adapun saran pada Tugas Akhir Rancang Bangun Sistem Monitoring Kecepatan Angin Berbasis Mikrokontroler Atmega32 pada Turbin Angin Tipe Horizontal Axis ini adalah sebagai berikut :

- a. Penggunaan disk rotary encoder sebaiknya menggunakan lebih banyak lubang, untuk meningkatkan nilai akurasi dari pembacaan sensor.
- b. Pembuatan mekanik untuk pembacaan kecepatan angin dapat diperbaiki dengan mengganti bahan plastik dengan aluminium dan direkatkan secara permanen.

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

- [1] A.T.Zhuga, "Design of Alternative Energy System: A Self-Starting Vertical Axis Wind Turbine for Stand-Alone Applications (charging batteries)," Chinhoyi University of Technology, Zimbabwe, 2005.
- [2] BMKG, "Stasiun Metrologi Maritim Tanjung Perak," Badan Meteorologi, Klimatologi, dan Geofisika, 2015. [Online].
- [3] BPP. (2012). Kajian Pembuatan SKEA (Sistem Konversi Energi Angin) di Sumatra Utara. Sumatra Utara: Pemerintah Provinsi Sumatra Utara.
- [4] Bukho, I. (2012). Retrieved Maret 27, 2016, from <http://digilib.polsri.ac.id/files/disk1/105/ssptpolsri-gdl-ilhambukho-5217-3-babii.pdf>
- [5] Kementrian Energi dan Sumber Daya Mineral Republik Indonesia, "Renstra," Direktorat Jenderal Ketenagalistrikan, Jakarta, 2015.
- [6] Octovhiana, K. D. (2013). Pendahuluan. *Cepat Mahir Visual Basic 6.0*, 1.
- [7] PT PLN (Persero), "Rencana Usaha Penyediaan Tenaga Listrik," PT PLN (Persero), Jakarta, 2013.
- [8] Saputro, H. (2012). Retrieved Maret 28, 2016, from www.akmi-batureja.ac.id
- [9] Sargolzaei, J. 2007. *Prediction of The Power Ratio in Wind Turbine Savonius Rotors Using Artifical Neural Networks*. Zahedan: Baluchestan University.
- [10] Setiono, D. (2011). Retrieved Maret 27, 2016, from [repository.usu.ac.id>bitstream](http://repository.usu.ac.id/bitstream)
- [11] UNY. (2011). Dasar Teori. Yogyakarta: Universitas Negeri Yogyakarta.
- [12] WHyPGen, "WHyPGen Wind Hybrid Power Generation Project," WHyPGen Wind Hybrid Power Generation Project, 25 June 2013. [Online]. Available: <http://whypgen-bppt.com/id/teknologi-whypgen/item/758-jenis-jenis-teknologi-energi-angin.html>. [Accessed 5 December 2015].

^[13] Wardhana, Lingga. 2006. Belajar Sendiri Mikrokontroler AVR Seri ATMega8535 Simulasi, Hardware dan Aplikasi. Penerbit: Andi Yogyakarta

BIODATA PENULIS



Sinta Dewi Rahmawati, biasa dipanggil Sinta merupakan anak pertama dari dua bersaudara. Dilahirkan di Surabaya, pada 27 Januari 1995. Menempuh pendidikan mulai dari SDN Bangah 1 kemudian melanjutkan pendidikan SMPN 2 Sedati, dan SMAS Antartika Sidoarjo. Pada tahun 2013 penulis melanjutkan pendidikan di prodi D3 Metrologi dan Instrumentasi, jurusan Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya. Penulis tercatat sebagai asisten di *Material Laboratory*, Teknik Fisika ITS. Apabila ada pertanyaan, kritik dan saran tentang Tugas Akhir ini, penulis dapat dihubungi melalui email sintashafa@gmail.com.